

**Universidad Carlos III de Madrid  
Escuela Politécnica Superior**



**Grado en Ingeniería de Sistemas Audiovisuales  
Trabajo Fin de Grado**

**Obtención, tratamiento y visualización de  
estadísticas mediante herramienta Web**

Autor: Oscar Leal Díaz  
Tutor: José Jesús García Rueda  
Directora: Myriam Escobar Zapardiel (Vodafone Spain)



# Agradecimientos

En primer lugar quiero agradecer el apoyo y la confianza depositada en mi, por parte de mi directora de proyecto, Myriam Escobar, porque sin su positividad y esfuerzo Analiz no habría salido nunca a la luz, y porque aunque la beca se acabara, sigue contando conmigo siempre que tiene oportunidad, gracias.

A Eva Nuño, por haber visto, en nuestro primer contacto, que tenía un hueco en su equipo. Por darme cariño incluso cuando el trabajo nos desbordaba a ambos.

A todo el equipo de formación y aprendizaje, en especial a Paco Franco, mi mentor en esto del PHP, en moodle, y en la ironía nivel experto.

Gracias a la confianza que me han dado mis compañeros, que ahora muchos son amigos, un becario en formación nunca fue un becario, sino uno más de la familia.

Mención especial a mis Padres y familia, que después de varios tropiezos, ven como sigo levantándome y peleando como me han enseñado desde pequeño.

A mis niñas: Aurora, María, Cris, Miri y Esther y a los top: Javi, Juanlu, Victor, Gijón, Bea, Raquel, Irene, Celia, Capi, Luisón, Abel y el Perroflauta, nunca una historia tuvo mejores protagonistas.

A @maribauty y @elegOde\_ por rescatarme y cuidarme en cada ocasión, incluso cuando soy incapaz de pedir ayuda.

A mi gente del grado: Jm, Pokipoki, Angel, Enol, Dmartin, Diemate, Mota, Carawapa, Nuria, María, Ade ... con risas y gente como vosotros, no necesitamos más para un día perfecto.

A Pacheco, Cris, Tincho y Ana, por formar parte de mi familia, no necesito veros para sentirlos cerca.

A Nacho, Ana, Txisis, Ardi, Laura, Piñi, Marta, David y Ro no se puede medir todo lo que me habéis dado en estos años, ojala pudierais entender, aunque fuera una parte muy pequeñita, todo lo que os necesito.

A Nona, que ha llegado hace poco, pero a base de mordisquitos ha sabido abrirse paso y ganarse todo nuestro amor.

Y por supuesto, gracias a la persona más importante que ha irrumpido en mi corazón. No existe frase que alcance a expresar todo el amor que siento por ti. Gracias por amarme, por apoyarme, por comprenderme, por salvarme. GRACIAS CHU. Tu Quieta Mala.



## Resumen

La finalidad de este proyecto es crear una solución capaz de recolectar datos acerca del uso de la herramienta Alejandría de la empresa Vodafone, procesarlos y mostrarlos al usuario de la forma más sencilla y autónoma posible. La solución desarrollada se denomina Analiz.

Este proyecto se realiza en el departamento de formación y aprendizaje, en la empresa Vodafone España, y surge como solución a la necesidad de conocer el número de accesos que sufre la herramienta Alejandría por parte de los teleoperadores.

Los agentes teleoperadores recurren a Alejandría para gestionar las llamadas que reciben de los clientes. Para Vodafone es importante conocer el flujo de visitas que recibe la aplicación para poder gestionar los 'planning' de carga de trabajo, así como para tener un seguimiento y ver si hay concordancia entre el uso de Alejandría y el número de llamadas que reciben los call centres de Vodafone.

Analiz es la respuesta a esta necesidad: combinando las mejores características de distintas aplicaciones, logramos una forma vistosa, simple y compacta de observar los datos estadísticos necesarios.

# **Abstract**

The goal of this project is creating a solution able to collect data from the Vodafone tool Alexandria, processing theme and display to the users in the easiest and self-assisted way.

This project has been made in the “Formación y Aprendizaje” area, to Vodafone Spain corporation, and come up as a response for the need of knowing the amount of access that Alexandria is been supporting from operators.

Operators turn to Alexandria to manage incoming calls from clients, to Vodafone is important know the number of visitors the application is having in order to plan workload, also they need to have the tracking and compare if there is concordance between Alexandria use and number of incoming calls in call centres.

Analiz is the answer to this need, it combines the best characteristics from different application, and the result is a brilliant, simple and compact way of watching statistics data.

## Índice

<b>Capítulo 1. Introducción.....</b>	<b>9</b>
1.1 Introducción.....	9
1.2 Ámbito.....	9
1.3 Motivación.....	10
1.4 Objetivos .....	10
1.5 Contenido de la memoria.....	11
<b>Capítulo 2. Estado del arte .....</b>	<b>13</b>
2.1 Introducción.....	13
2.2 Infoguía.....	13
2.3 Estructura de la web .....	14
2.4 Estructura del servidor.....	15
2.5 Análisis del sistema anterior.....	16
2.5.1 Recolección de datos.....	16
2.5.2 Procesado de datos.....	17
2.5.3 Visualización de estadísticas.....	18
2.5.4 Almacenado de historiales .....	19
2.5.5 Información reportada.....	19
2.6 Puntos positivos y negativos.....	20
2.7 Conclusión.....	21
<b>Capítulo 3. Historia del Proyecto .....</b>	<b>22</b>
3.1 Introducción.....	22
3.2 Alejandría .....	22
3.3 Planteamiento.....	24
3.4 Carácter de la primera reunión.....	24
<b>Capítulo 4. Implementación .....</b>	<b>26</b>
4.1 Introducción.....	26
4.2 Funcionalidad y Requisitos.....	26
4.2.1 Funcionalidad .....	26
4.2.2 Requisitos .....	27
4.3 Recolección de datos .....	27
4.3.1 Sistema anterior.....	28
4.3.2 Alternativas.....	28
4.3.3 Solución Final.....	28
4.4 Procesado de datos.....	29
4.4.1 Sistema anterior.....	29
4.4.2 Alternativas.....	29
4.4.3 Solución Final.....	30
4.5 Almacenamiento de datos .....	30
4.5.1 Sistema anterior.....	30
4.5.2 Alternativas.....	31
4.5.3 Solución final.....	32
4.6 Visualización de datos.....	35
4.6.1 Sistema anterior.....	36
4.6.2 Alternativas.....	36
4.6.3 Solución final.....	37
4.6.3.1 Entorno Web.....	37

4.6.3.2 FusionCharts.....	42
4.7 Documentación .....	44
<b>Capítulo 5. Implantación y pruebas .....</b>	<b>45</b>
5.1 Implantación .....	45
5.2 Pruebas iniciales.....	45
<b>Capítulo 6. Planificación y Presupuesto .....</b>	<b>47</b>
6.1 Planificación .....	47
6.2 Presupuesto.....	50
6.2.1 Coste Personal .....	50
6.2.2 Coste Software .....	50
6.2.3 Coste Hardware.....	51
6.2.4 Resumen de costes.....	51
<b>Capítulo 7. Líneas Futuras y Conclusiones .....</b>	<b>53</b>
7.1 Líneas Futuras .....	53
7.1.1 Automatización .....	53
7.1.2 Visionado de estadísticas .....	53
7.1.3 Perfiles .....	54
7.1.4 Optimización .....	54
7.2 Conclusiones .....	54
<b>Capítulo 8. Bibliografía .....</b>	<b>56</b>



# Capítulo 1. Introducción

---

## 1.1 Introducción

La forma más enriquecedora que tenemos de medir cualquier elemento se obtiene mediante la estadística.

La estadística es la ciencia formal dedicada al estudio de una característica en una población, recogiendo datos, organizándolos, representándolos y analizándolos para sacar conclusiones acerca de dicha población.

En el ámbito empresarial, debemos averiguar si vamos en la dirección correcta y si los cambios que hacemos tienen verdadero impacto, para ello recurrimos a la recolección de datos. Dichos datos se agrupan por *KPIs (Key Performance Indicator)*[1], se analizan y se sacan conclusiones acerca de las decisiones tomadas.

El marco del proyecto se da justo en ese punto, la recolección y agrupación de datos para posteriormente ser capaces de analizarlos.

En el proyecto se da una solución ‘*End to End*’ para el procesado y visualización de accesos al catálogo web con el que cuentan los tele-operadores de Vodafone. Dicha solución es un conjunto de herramientas que permiten un manejo fácil y cómodo del sistema. Analiz, la herramienta web creada en el proyecto, es la encargada de la visualización de los datos, de cara a reportar el flujo de visitas de las páginas internas de Vodafone consultadas por los agentes.

Con este proyecto intentamos que la obtención y actualización semanal de las estadísticas deje de ser un trabajo tedioso, para convertirse en algo mucho más sencillo, y flexible, resultando un sistema casi autónomo que libere a los trabajadores de dicha tarea. De esta forma optimizamos el tiempo, y dosificamos los recursos humanos disponibles.

Así mismo, es importante poner nuestra atención en la mejora de visualización de las estadísticas. Debemos conseguir un sistema que tenga una interfaz más simple y sencilla, de forma que los datos relevantes sobresalgan del resto.

## 1.2 Ámbito

La realización del proyecto se ha dado por completo en la empresa Vodafone España, en el departamento de Formación y Aprendizaje.

Vodafone es una empresa de nivel internacional, del entorno de las telecomunicaciones. Es la 2ª empresa más poderosa de España en cuanto a telefonía móvil se refiere, con más de 17 millones de clientes[2].

El departamento de Formación y Aprendizaje es el responsable de la formación que reciben los tele-operadores, así como de los soportes web donde se aloja la información que los clientes les requieren en sus llamadas telefónicas. De todas las tareas que conforman este departamento, una de las más solicitadas es la de *reporting*.

Se reportan todo tipo de datos relativos al rendimiento de los tele-operadores. Y en ese punto es donde éste proyecto tiene cabida.

### 1.3 Motivación

A la llegada del autor de este proyecto a la beca de colaboración con Vodafone, se sumó el despliegue de una nueva herramienta para los tele-operadores: Alejandría.

Alejandría es una plataforma web que introduce un cambio de filosofía a la hora de atender llamadas. Con la implantación de ésta herramienta se hacía necesaria la recolección de datos relativos a las páginas visitadas por los agentes. Hasta ese momento esto mismo se hacía con la antigua herramienta: Infoguía.

No se conocía el modelo que había que seguir a la hora de consultar dichas estadísticas, sin embargo se tenía claro que había que huir del modelo anterior. Con una estructura muy recargada, y arrojando demasiada información innecesaria, el cuadro de mando que se conformaba para Infoguía estaba obsoleto. Por ello, surge la necesidad de buscar una nueva forma de mostrar esos datos recopilados, que referencian el uso que se hace de la nueva herramienta Alejandría.

Es cierto que el cambio de herramienta de cara a los agentes no hacía obligatoria la modificación de nuestras herramientas de *reporting*, sin embargo, ese punto de inflexión era motivo suficiente para justificar un cambio significativo a la hora de recopilar y mostrar los datos. Además, buscamos una mayor flexibilidad de sistema, así como una mejora en los tiempos de obtención de los mismos. En resumidas cuentas, buscamos una optimización del sistema completo.

### 1.4 Objetivos

El principal objetivo del proyecto es: crear un proceso '*End to End*', para la recolección y representación de los datos estadísticos de los accesos a Alejandría.

Dicho objetivo podemos segmentarlo en varios:

- Estudiar la comunicación servidor – cliente mediante servidor ftp. Para la obtención de los archivos *logs*, que son suministrados en un servidor remoto.

- Definición y configuración de un estándar de estructura para los archivos *logs*, para el posterior procesamiento de los mismos.
- Estudiar la mejor solución para el agrupamiento de *logs* y extracción de parámetros comunes.
- Estudiar e implementar una base de datos de manera óptima. Dicha base de datos será a la que se recurrirá para el almacenamiento de los datos obtenidos semanalmente.
- Creación de un soporte desde el que obtendremos la agrupación de los datos. En ella seremos capaces de consultar las estadísticas que nos ofrece el departamento semanalmente. Deberá comunicarse con la base de datos.
- Persistencia de datos a lo largo del tiempo, con un visionado máximo de 1 año y 3 meses anterior a la fecha del mes en curso.

### 1.5 Contenido de la memoria

A lo largo de la memoria encontraremos la descripción detallada y en profundidad de todo el proyecto llevado a cabo. Esta memoria se ha estructurado en capítulos.

En el primer capítulo encontramos la introducción, el ámbito en el que se desarrolló el proyecto y las motivaciones y objetivos que lo componen.

Posteriormente, en el segundo capítulo, el estado del arte nos describe la tecnología que se usaba antes de éste proyecto en el departamento. Tanto la web de información que usaban los agentes de Vodafone, como el sistema que se usaba para obtener los datos. Analizaremos y describiremos los pasos que se llevaban a cabo para obtener las estadísticas. Esto nos servirá para entender mejor las opciones que había para desarrollar el proyecto.

En el tercer capítulo, historia del proyecto, hablaremos del cambio que supone Alejandría, del planteamiento de la nueva aplicación, Analiz, y de la importancia de la primera reunión con la jefa del proyecto: Myriam Escobar.

A continuación, en el capítulo cuarto, abordaremos toda la implementación del proyecto, analizando cada una de las partes que lo componen, estudiando las soluciones alternativas que se hubieran podido tomar y justificando las escogidas, previo paso por la funcionalidad y requisitos del sistema.

En el capítulo quinto, detallamos la fase de implementación de la herramienta, así como las pruebas llevadas a cabo.

Posteriormente, en el capítulo sexto, abordamos la planificación del proyecto. Podremos observar los tiempos específicos que ha llevado el desarrollo de Analiz. Así mismo un detallado resumen de gastos nos dará el presupuesto que se ha empleado para la elaboración de la herramienta.

Por último, en el capítulo siete, sugerimos las posibles líneas de mejora que puede seguir el proyecto. Observando las partes más débiles del mismo, y

observando las necesidades del departamento. Además, terminamos la memoria con las conclusiones del proyecto.

## Capítulo 2. Estado del arte

---

### 2.1 Introducción

En éste capítulo analizaremos el sistema que se estaba usando en el departamento de información y aprendizaje para la obtención de estadísticas en torno a la página de información Infoguía.

Haremos un repaso de las soluciones adoptadas con una visión detallada, para comprender el proceso que se llevaba a cabo, y de esta manera, ser capaces de llegar a una futura solución óptima, tal y como se hizo para desarrollar el proyecto.

### 2.2 Infoguía

Infoguía[3] es el sistema de información que usan los agentes para atender las llamadas de los clientes de Vodafone. Se trata de una herramienta web donde se aloja todo el contenido sobre el que los usuarios pueden solicitar información.

El contenido de Infoguía se actualiza de manera manual diariamente, a petición de los distintos departamentos. Este proceso se lleva a cabo en el departamento de formación y aprendizaje. Las solicitudes se generan mediante email, y las peticiones de cambios se efectúan mediante una cola de prioridades.

El trabajo a realizar se basa en adaptar los contenidos que llegan por email a formato HTML, para que, de forma unificada, se muestre dentro de Infoguía. Para ello existen plantillas HTML que se rellenan con la información necesaria.

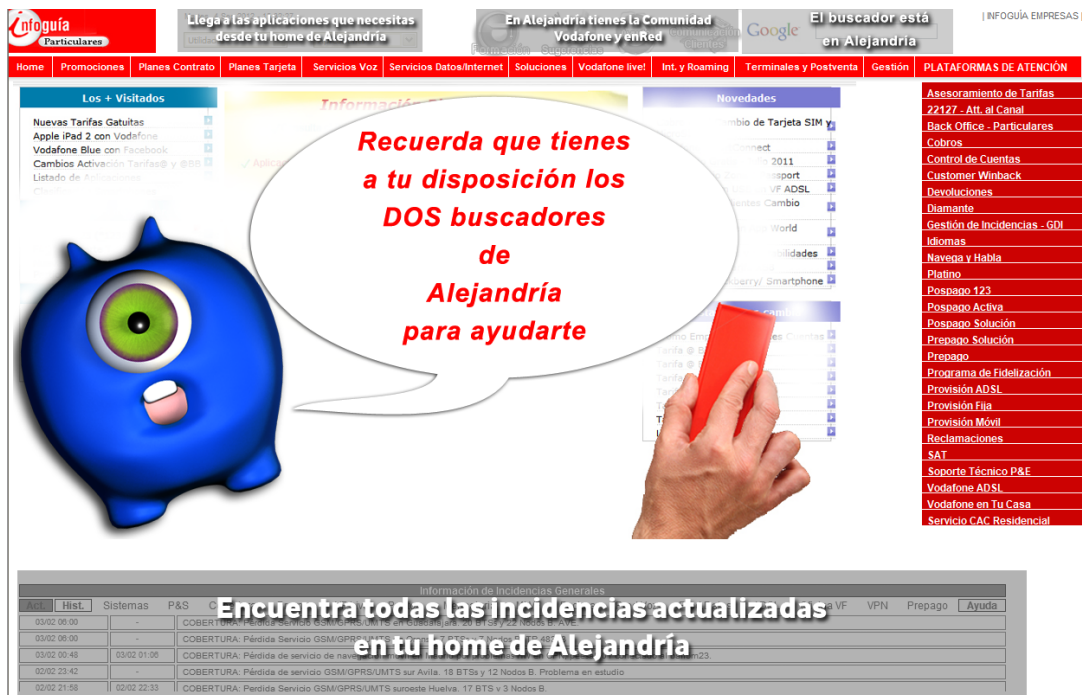


Figura 2.1 Infogúa

## 2.3 Estructura de la web

La web Infogúa se divide en dos *Homes*, o accesos principales: Particulares y Empresas. Los accesos a cada una de ella son independientes, se tratan de sitios web diferentes.

La filosofía que envuelve Infogúa define su estructuración web. La información se puede categorizar, salvo contadas excepciones, en 2: productos y servicios. Todo lo que un agente debe decirle a una persona que llama, debe estar en esa información. La forma de llegar a ella se da mediante páginas intermedias que sirven para categorizar los contenidos: si hablamos de prepago o postpago, o si hablamos de promociones puntuales o de servicios ofrecidos por Vodafone, todo ellos nos lleva a esas fichas donde localizamos los datos descriptivos del producto.

Además de estos productos y servicios, desde Infogúa podemos acceder a otras aplicaciones web que pueden necesitar los agentes como por ejemplo mapa de cobertura o listado de incidencias. Pero este contenido, al estar alojado fuera de los servidores a los que tenemos acceso, no podremos analizar su flujo de visitas. Por lo tanto carecen de interés para el proyecto.

## Planes de Precios y Tarifas Cliente Contrato

Consulta los **SMS** que reciben los Clientes al realizar un cambio de Plan de Precios

Consulta los **Códigos y Descuentos de los planes de Precios**

Consulta el **Argumentario** para clientes que han recibido un sms para indicarle que su plan deja de estar comercializable

Los Mejores Planes Voz	Planes Para Hablar y Navegar	Planes de Datos	Recomendador Planes
<b>Planes Tarifas por Tallas/A Mi Medida</b> Tarifa XL Clásica Tarifa L Clásica Tarifa M Tarifa S  <b>Planes XS</b> Tarifa XS Simple Tarifa XS 90x1 a Todos Tarifa XS 90x1 24h Tarifa XS Mensajes Gratis Tarifa XS Fines de Semana Tarifa XS Mi País Tarifa XS6 Tarifa XS8  <b>Planes Valor Medio</b> Contrato Super 90x1  <b>Otros planes (por petición Cliente)</b> Plan 8	<b>Planes Tarifas por Tallas/A Mi Medida@</b> Tarifa @XL Clásica Tarifa @L Clásica Tarifa @M Tarifa @M Premium Tarifa @S Tarifa @XS  <b>Planes Tarifas por Tallas/A Mi Medida @ BlackBerry</b> Tarifa @XL Clásica (BlackBerry) Tarifa @L Clásica (BlackBerry) Tarifa @M (BlackBerry) Tarifa @S (BlackBerry) Tarifa @XS (BlackBerry)  <b>Solo si el cliente lo solicita</b> Tarifa @L+ Clásica Tarifa @M+ Clásica Tarifa @S+ Clásica	<b>Para Navegar con el móvil</b> Tarifa Plana Internet en el Móvil Tarifa Plana Internet en el Móvil Plus Tarifa Datos por Defecto Tarifa Plana Live! por conexión Tarifa Plana Live! mensual Tarifa Navega en PDA Plus Tarifa Plana Internet SE  <b>Internet</b> Internet Contigo Deberes Internet Móvil 500MB Internet Móvil 1 GB  <b>Internet para Tablets</b> Internet Contigo Ilimitado Tablets Internet Contigo Express Tablets Internet Contigo Oro Tablets Internet Contigo 15 Tablets  <b>Tablas Resumen</b>	<b>Asesor de Tarifas Web</b> Asesor de Tarifas (si no funciona Web) Recomendador Datos Contrato Comparación Competencia  <b>Tarifas n° especiales</b> Servicios 700 y 900 Llamadas a 800 (STA y STE) Llamadas a 905 (STA ) Urgencias e Información  <b>Utilidades</b> Festivos Nacionales 2011  <b>Otros enlaces</b> ★ Clasificación Smartphones Política de Seguridad Planes Precio Tarjeta

Figura 2.2 Infoguía Home

## 2.4 Estructura del servidor

Infoguía se encuentra alojada en dos servidores distintos, en dos máquinas distintas. Toda la afluencia de visitas que recibe la página se cataliza, aproximadamente, al 50% entre los dos servidores, de esta forma se evita una posible saturación en la red sobre uno de ellos.

De cara al personal de Vodafone que se dedica al mantenimiento de la web, tenemos 3 entornos de trabajo:

- Desarrollo: se trata de una replica del sitio completo, en la cual se realizan pruebas antes de llegar a una versión definitiva. Esta diseñada para que sólo el personal del departamento vea el contenido.
- Preproducción: en este caso es otra réplica del contenido original que contiene Infoguía. En preproducción se introduce todo el contenido que está preparado para subir a la web, para que el departamento que encargó dicho contenido pueda hacer las comprobaciones necesarias.
- Producción: es el entorno de trabajo previo a subir el contenido que hay en la web. Cada día, hay 3 *scripts* de copiado, que suben toda la información disponible en este entorno a los servidores públicos.

Al ser tres entornos diferentes de trabajo se requiere una precisa coordinación, para que en los tres haya la última información disponible, de otro modo podríamos estar modificando un archivo en desuso o con información atrasada.

Además de los 3 entornos de trabajo antes mencionados, tenemos un archivo de históricos donde se van metiendo los archivos antes de las modificaciones, de

esta forma, si hay que volver a una versión anterior, la tenemos catalogada por fecha y se recupera al instante.

## 2.5 Análisis del sistema existente

Necesitamos conocer el sistema que se usaba anteriormente al proyecto para comprender los cambios que se han realizado, conocer las limitaciones del mismo, y saber el punto del partida en el que nos encontramos.

Los datos estadísticos de Infoguía se obtienen semanalmente, así pues, a principio de semana se deben obtener los datos relativos a la semana anterior.

En la siguiente figura podemos observar el proceso de programas por el que debe pasar.



Figura 2.3 Flujo de programas del sistema anterior

### 2.5.1 Recolección de datos

Los datos que debemos recopilar, se encuentran en un servidor remoto, al que solo tenemos acceso mediante un cliente FTP (Filezilla[4]) autorizado por el departamento de seguridad de Vodafone. Por supuesto necesitamos un nombre de usuario y contraseña para conectarnos al mismo. Una vez estamos dentro del servidor, debemos copiarnos a cualquier carpeta local los archivos autogenerados a lo largo de la última semana. Necesitamos 7 archivos, uno para cada día de la semana, que se suelen generar entorno a la 1 de la madrugada, y que contiene los datos del día anterior. Así pues, si por ejemplo queremos obtener los datos que contiene la semana 36 que contiene los días del 03/09/2012 al 09/09/2012 deberemos coger los archivos cuyo nombre es *Access.201209030000* al *Access.201209090000*. Como detalle cabe remarcar que los archivos autogenerados, no se generan siempre a la misma hora, pudiéndose retrasar varias horas, así mismo tampoco se generan siempre con el mismo nombre, a veces tenemos derivaciones en los últimos dígitos del mismo. Además, ocasionalmente, los archivos ni siquiera se generan, debiendo hacer la pertinente reclamación para que el departamento correspondiente los recupere y los mande.

Esto se debe realizar con los 7 archivos de cada servidor. Cómo observamos, anteriormente se ha explicado que hay 2 servidores que se reparten el flujo de visitas. Si bien aproximadamente se reparte en un 50% a cada servidor, esta



medida no es exacta, por lo tanto necesitamos recuperar los 7 archivos de cada servidor para tener una medida más estricta.

Cada archivo tiene aproximadamente una extensión de 50 mB, si es un día de diario, o de 10 mB para los festivos y fines de semana, dado que tienen menor carga de llamadas los tele-operadores, por lo tanto acceden menos a la herramienta.

El copiado de datos por medio de Filezilla, lleva aproximadamente un tiempo total de 10 minutos.

Los archivos contienen una a una cada página que se ha visitado de Infoguía. Su URL, la fecha y hora en que fue visitada, así como la respuesta que recibimos del servidor. Esto último es imprescindible para conocer si la página visitada realmente funciona o genera algún fallo, bien sea porque el agente escribe mal la dirección, o bien porque hemos eliminado alguna de las páginas de contenido, y por lo tanto cuando intentamos acceder a ella generamos un error. En cualquier caso, tener el dato nos permite un mejor control de errores.

### 2.5.2 Procesado de datos

Una vez tenemos los 14 archivos recuperados, y copiados en local, debemos procesarlos.

El procesado consiste en agrupar los 14 archivos y categorizarlos en 2 archivos tipo 'txt'. Estos dos archivos serán la agrupación de los links validos por un lado, y los links erróneos por otro. Los links validos son el resultado de un acceso a la web correcto, con respuesta 200 por parte del servidor. Por contrario, los links erróneos, son los que reciben otro tipo de respuestas por parte del servidor, y significa que la página web solicitada no ha sido posible visitar.

Una vez tenemos estos dos archivos, generados por una macro de Excel, recurrimos a Business Object para categorizar los diferentes links correctos, y gestionar la información. Ésta consulta lo que hace es agrupar las entradas analizando su URL, para contabilizar el número de visitas a zonas de la web concretas.

La categorización que acabamos de realizar queda plasmada en un archivo con extensión 'txt', que es muy sencillo de tratar, y tiene menos peso que otras extensiones. Así pues, el archivo resultante debe contener información de acceso según la siguiente agrupación:

- Home principal
- Productos
- Servicios
- Páginas intermedias
- Agrupaciones secundarias
- Otros accesos

El siguiente paso que debemos seguir es copiar el contenido a una hoja de Excel, llamada cuadro de mando, que recoge esta y mucha otra información que nos interesa analizar en el departamento de Formación y Aprendizaje.

Anteriormente se copiaba la información dato a dato del archivo al cuadro de mando. Este tedioso proceso fue sustituido por una macro en Excel, que era la encargada de copiar esos datos al cuadro de mando, para posteriormente poder visualizarlos.

### 2.5.3 Visualización de estadísticas

Para la visualización de las estadísticas obtenidas de Infoguía, recurrimos al cuadro de mando. Se trata de una hoja de calculo Excel con los datos agrupados, que muestran el número de accesos semanales a lo largo de un año fiscal.

Dicho cuadro de mando es un documento bastante complejo de manejar, dado que contiene más de 120 columnas, y 52 semanas. Así mismo, contiene varias hojas donde se rescatan diferentes categorías de datos. Todo esto se plasma en gráficas que hacen un poco más simple el visionado, pero algo menos riguroso y por supuesto, menos exhaustivo.

En la siguiente figura podemos observar el cuadro de mando que se usa para la visualización de estadísticas. Por motivos de seguridad de Vodafone, no podemos incluir imágenes con mejor resolución, así mismo, los datos que aparecen son totalmente ficticios.

MESES CONTENIDO Y CONTENIDOS											HOMES GENERALES																																																																																																																																																																																																																																																																																																																																																																																																																																																																																						
		3		4		5		6		7		8		9		10		11		12		13		14		15		16		17		18		19		20		21		22		23		24		25		26		27		28																																																																																																																																																																																																																																																																																																																																																																																																																																													
		OPERATIVA		OPERATIVA		COMERCIAL		COMERCIAL		OPERATIVA		COMERCIAL		COMERCIAL		COMERCIAL		COMERCIAL		TOTAL VISITAS HOMES GENERALES		TOTAL VISITAS HOMES 5 y 6 y 7		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SERV. ATT. PART. INHAB. PLAN		SER	

### 2.5.4 Almacenado de historiales

Una vez hemos terminado con todo el procesado de datos, es necesario realizar una copia de seguridad de los archivos que hemos obtenidos. De esta forma garantizamos la seguridad de información, por si en un futuro hay algún fallo en los discos duros.

Simplemente debemos copiar los archivos generados a unos directorios remotos que nos ha proporcionado el departamento de seguridad, y de los que nos garantizan que si por algún fallo se rompen, son capaces de recuperar toda la información, cosa que con los discos duros locales no nos aseguran.

De forma adicional, los datos guardados en el histórico nos permiten recurrir a ellos, por si buscamos un dato más concreto, que se pierda al hacer el agregado y guardar en el cuadro de mando.

### 2.5.5 Información reportada

Los datos que más se suelen consultar del cuadro de mando respecto a Infoguía son:

- **Accesos totales:** Se lleva un seguimiento de cual ha sido la afluencia de visitas al conjunto de la web. Así mismo se hace especial atención en los valores disparados, tanto por altos como por bajos, y se busca una posible justificación a los mismos, recurriendo, si es necesario, a otros departamentos como tecnología, para comprobar que no ha habido caídas en los sistemas. O productos, por si se ha ofertado algún producto que pudiera estar resultando muy efectivo.
- **Accesos a productos:** Se consulta el acceso a productos, para comprobar que realmente hay productos que se siguen consultando o si ha funcionado alguna campaña específica.

Por supuesto podemos reportar otros datos según sean requeridos. En realidad se arrojan muchos más datos de los que al final son utilizados. No se lleva un seguimiento tan férreo del uso de Infoguía como el cuadro de mando sugiere. Sin embargo esto no está del todo mal, por si en algún momento hiciera falta recurrir a ellos, es bueno que ya estén procesados y listos para ser visionados. Sin embargo esto hace que el proceso sea mucho más lento.

Hay dos formas de reportar la información que obtenemos:

- **Vía email:** Si alguna persona en concreto, necesita conocer algún dato, simplemente lo recogemos y mandamos vía email. Es un proceso manual, por lo tanto, poco deseado. Por esta razón se usa en contadas ocasiones.
- **Compartir el cuadro de mando:** la otra opción que tenemos para ofrecer nuestros datos es compartiendo el cuadro de mando. El archivo está colocado en una carpeta compartida, de modo que al pedir un dato,

nosotros damos acceso a esa persona, y a partir de ese momento puede entrar a consultarlo en cualquier momento, copiarlo y revisarlo las veces que necesite. Obviamente, el cuadro de mando está protegido con contraseña, de forma que sólo las personas autorizadas podrán modificar su contenido.

## 2.6 Puntos positivos y negativos

Hemos visto el sistema que se usaba anteriormente para recoger los datos estadísticos que necesitamos. Ahora recurriremos a un análisis de los puntos positivos y negativos, para tener una visión más sintetizada y observar las partes del proyecto que se podrían reutilizar.

<b>PUNTOS POSITIVOS</b>	
<b>Gestor de archivos FTP</b>	A parte de ser una imposición corporativa, se trata de una solución sencilla y sin ningún tipo de desarrollo asociado. Fácil de usar.
<b>Excel como gestor y procesador de datos</b>	A pesar de utilizar Visual Basic, un lenguaje de programación algo antiguo, Excel es una herramienta que poseen todos los equipos.
<b>Excel como cuadro de mando</b>	La mayoría de trabajadores conocen y manejan a un nivel medio alto MS Excel, por lo tanto no necesitan periodo de adaptación, y son bastante independientes.

Tabla 2.1 Puntos Positivos del Sistema Anterior

<b>PUNTOS NEGATIVOS</b>	
<b>Solución fragmentada</b>	La obtención de datos requiere demasiada atención personal, al tratarse de procesos separados.
<b>Tiempo procesado</b>	El tratamiento de los datos requiere demasiado tiempo de procesado.
<b>Exceso de datos mostrados</b>	Se visualizan demasiados datos inútiles, lo que hacen menos claros los que realmente son útiles
<b>Compartición de datos</b>	Compartir los datos del cuadro de mando se hace de manera manual, por lo tanto necesitas tiempo y a una persona que se dedique a ello cuando sea requerido.

Tabla 2.2 Puntos Negativos del Sistema Anterior

## 2.7 Conclusión

Con el despliegue de la nueva herramienta Alejandría, podemos plantear un cambio en la gestión de las estadísticas. Si bien es cierto que el sistema anterior funcionaría perfectamente con Alejandría, es un buen momento para introducir mejoras, y sobre todo, un lavado de imagen al *reporting* de estadísticas.

Con el tiempo justo con el que contamos, sería indicado aprovechar parte del material del que disponemos, sin cometer los mismos errores, y aportando valor al trabajo que ya hay creado.

Debemos focalizar nuestras acciones en la mejora de visualización de las estadísticas, pero sobre todo en la automatización de los procesos. Esto librerá de carga a los trabajadores, que podrán emplear su tiempo en otras actividades, logrando así un mayor rendimiento del equipo de formación y aprendizaje.

## Capítulo 3. Historia del Proyecto

---

### 3.1 Introducción

La realización del proyecto busca la implementación de un sistema completo que de una solución a la recolección, procesado y visionado de estadísticas procedentes de Alejandría. Nos basaremos en la versión anterior que trata los datos de Infoguía. Sin embargo, para llegar a una solución optima replantearemos todo lo que ya hay construido. Es necesario revisar cada punto, pensar en alternativas y tomar decisiones de diseño diferentes si queremos obtener una mejora respecto a la versión anterior.

El proyecto, antes de tener forma, tiene nombre, se llama Analiz. El nombre viene, como es obvio de Analizar, y creemos que es un buen nombre para una herramienta dedicada a las estadísticas. Como vemos a continuación, el logotipo es un juego de imágenes entre un ojo haciendo las veces de 'A' para crear la palabra Analiz. El sentido del ojo es remarcar la finalidad de la herramienta, que es llevar el seguimiento de accesos a Alejandría.



Figura 3.1 Logo Analiz

### 3.2 Alejandría

Dado que el proyecto se implanta para Alejandría, la nueva herramienta web que usan los tele-operadores, se hace necesaria una pequeña descripción de la misma.

El proyecto se plantea con el cambio de plataforma web a la que recurren los tele-operadores para responder a las llamadas de los clientes de Vodafone.

El cambio de Infoguía a Alejandría supone un cambio de filosofía en la resolución de llamadas. La diferencia radica en que: en Infoguía toda la información se encuentra en las fichas de producto y servicio, por lo tanto debemos conocer la naturaleza de la llamada para poder atenderla. Sin embargo Alejandría basa su funcionamiento en procesos, cada llamada se puede atender con un proceso asociado simplemente siguiendo un árbol de decisiones basadas en dos respuestas: Sí o NO. Así pues, una vez detectada la necesidad inicial, conseguimos que mediante preguntas y respuestas cortas todos los clientes reciban la misma solución al mismo problema.

Alejandría no necesita que los agentes conozcan donde se aloja la información final, simplemente necesitan conocer el motivo de la llamada y seguir el proceso que aparece. Esto hace que los agentes precisen recordar menos información, y que den una respuesta unificada ante la misma casuística.

Es un sistema sencillo, que requiere muy poco tiempo de aprendizaje, esto es sin duda una gran ventaja ante la inmensa rotación de personal que tienen los 'call centers.'

Cada grupo de trabajo de Vodafone, tiene su propia 'Home', es decir, una web donde se recogen todos los procesos por los que un usuario puede llamar para pedir información. Dentro de cada 'Home' encontramos los diferentes procesos referidos a grupo de clientes.

En la siguiente figura podemos ver la página principal de Alejandría donde se encuentran alojadas todas las 'Homes'.



Figura 3.2 Página Principal de Alejandría

En la siguiente figura podemos observar la situación final a la que hemos llegado tras seguir el árbol de decisión de un proceso concreto.

Al margen de los procesos, la información sobre productos se puede encontrar en la biblioteca de archivos, una de las partes más importantes del proyecto de Alejandría. En dicha biblioteca se almacena toda la información categorizada por clientes y tipo de archivo que es necesaria para la labor de los agentes. Podemos encontrar más de 1000 archivos, que recogen todo el conocimiento acerca de los productos vigentes y en periodo de extinción de Vodafone.

Otro de los pilares de Alejandría son los documentos de agente. Dichos documentos tienen información de carácter privado necesaria para que los agentes sean capaces de tipificar incidencias, o modificar en los sistemas las peticiones que los usuarios necesiten, bien sean cambios de precios, recobros, o cualquier incidencia que puedan reportar.

### Cobro de conexiones de datos no realizadas

ParticularesPospago

Fecha de creación: 13/08/2012

Última modificación: 13/08/2012

**Descripción del proceso:**  
Este procedimiento describe la manera de actuar ante llamadas de clientes que indican que quieren darse de baja de una tarifa de Internet en el móvil o realizar un cambio de tarifa de Internet en el móvil.

**Modo Texto**

PASO 1

Comprueba si tiene conexiones facturadas

SI

PASO 2

Comprueba si tiene alguna tarifa activa en su línea

NO

**COMPRUEBA:** Comprueba si tiene alguna tarifa activa en su línea o no. Para ello accede a sistema/Cuaderno de servicio/GPRS.

¿Tiene tarifa @ ó tarifa plana de internet contratada?

SI NO

PASO 3

Ofrece TP de datos ó @

**DI:** Ofrece al cliente darse de alta en una tarifa plana de internet o una Tarifa @. Apóyate en el Recomendador de Tarifas y en su defecto aconseja al cliente en base a sus necesidades.

Siguiente Paso

Figura 3.3 Ejemplo de Proceso

### 3.3 Planteamiento

Se requiere una solución para el procesado y visualización de datos procedentes de la herramienta Alejandría. Puede realizarse basándose en la herramienta anterior, o crear un sistema totalmente nuevo.

Es importante que Analiz requiera poco mantenimiento, y sea de fácil actualización en caso de ser necesario. Así mismo, el procesado de datos debe ser suficientemente autónomo para no cargar de trabajo al departamento.

### 3.4 Carácter de la primera reunión

La primera reunión con la Jefa de departamento, y tutora en el proyecto, Myriam Escobar, es de alta importancia. En ella se sientan las bases del trabajo a



realizar. No se realizaran especificaciones muy concretas, sin embargo, se da un sentido completo al trabajo.

En este primer acercamiento no se imponen plazos fijos, sino que se hace un esbozo del camino que seguirá el proyecto. Además, es una reunión primordial para comprender la necesidad por la que se inicia el proyecto, y lo que se espera de él.

## Capítulo 4. Implementación

---

### 4.1 Introducción

Este capítulo es el eje principal del proyecto. En él, analizaremos una a una las decisiones tomadas. Justificando las elecciones realizadas, estudiando los puntos positivos y negativos de cada opción.

Si bien el análisis se hace desglosando el proyecto punto por punto, es importante mantener una visión global del mismo para no perder de vista la perspectiva, y cumplir con los objetivos marcados.

### 4.2 Funcionalidad y Requisitos

Como primer paso para realizar el proyecto, es necesario un planteamiento del mismo. Analizamos el sistema anterior, para tener claro las características, y acciones que podemos reutilizar, y aquellos errores en los que no queremos volver a incurrir. De esta forma nos aseguramos de que todo el equipo está alineado, y todas las ideas que surjan van en la misma dirección. No se trata de un análisis de las soluciones adoptadas anteriormente, y si se pueden remplazar por otras, se trata de un estudio grosso modo de la totalidad del proyecto. Una foto que nos dé la imagen completa de la situación del mismo, y las posibilidades existentes.

#### 4.2.1 Funcionalidad

Una vez tenemos la visión global del sistema, ajustamos los objetivos del mismo. Qué queremos obtener, y de qué forma. En este punto, aparece la principal característica del proyecto: automatización. No queremos tener a una persona demasiado tiempo obteniendo estas estadísticas. La idea es conseguir las empleando a una persona el mínimo tiempo posible. El proceso si puede llevarse el tiempo necesario, sin embargo, no podemos tener a una persona dedicada a estas estadísticas, necesitamos a provechar los pocos recursos humanos de los que disponemos. Por esto, el sistema ha de ser suficientemente autónomo. Sin duda será necesaria una persona la que puntualmente lo ponga en marcha, pero al margen de esto, no debería requerir demasiada dedicación.

El siguiente objetivo que buscamos es la simplicidad, siguiendo una de las premisas que gobierna Vodafone : *'simplicity'*. Esto hace referencia a conseguir el mejor resultado de la forma más sencilla posible. Por eso mismo, a la hora de la presentación de resultados, queremos huir del anterior cuadro de mando que reportaba el departamento. Buscamos una forma simple y fácil de entender para visualizar los resultados. Así mismo, queremos que el sistema sea flexible, ofreciendo la posibilidad de obtener datos más generales o específicos según sea necesario.

Otra de las características del sistema, debe ser que sea de fácil modificación. Es decir, que no requiera demasiado desarrollo para introducir algún dato nuevo que queramos añadir. De esta forma facilitamos que cualquier persona del equipo pueda realizar modificaciones siempre que sea necesario. En cualquier caso, se hará necesaria la creación de un manual, para entender la herramienta y poder realizar los cambios sin necesidad de estudiarla a fondo.

El sistema creado debe tener gran memoria de histórico, de entre 6 meses y 1 año, a partir de ahí, toda la información puede quedarse en el sistema, o almacenarse en un histórico a parte.

#### 4.2.2 Requisitos

En la siguiente tabla podemos observar los requisitos que debe cumplir el proyecto, cumpliendo con las funcionalidades anteriormente descritas.

<b>PROCESO AUTOMÁTICO</b>
La obtención de estadísticas debe ocupar poco tiempo de los trabajadores. Se establece un tiempo valido de 1 hora por semana para la extracción de las mismas.
<b>SIMPLICIDAD</b>
Es necesario que el sistema sea simple a la hora de obtener la estadísticas, que muestre los datos de una forma clara y concisa, destacando los datos relevantes frente a los que no lo son tanto.
<b>GRAN CAPACIDAD</b>
Debe mostrar datos anteriores a 6 meses, y con un máximo de 12 meses. Los datos descartados deben almacenarse en un archivo histórico.
<b>ACTUALIZACIÓN</b>
Si en algún momento la aplicación necesita actualización, deber ser sencilla, y contar con un manual de usuario con instrucciones específicas.
<b>PRIVACIDAD</b>
Los datos deben permanecer inmutables para los usuarios ajenos al departamento, de forma que mantengan su validez.

Tabla 4.2 Requisitos

#### 4.3 Recolección de datos

Debemos obtener los archivos de log generados en los servidores. Los nuevos servidores, los que alojan Alejandría, funcionan de un modo similar a lo que vimos de Infoguía. Tenemos dos servidores, que reparten el peso de las visitas al 50%. Por lo tanto, cada uno de los servidores genera un archivo log con la fecha del día anterior. Del mismo modo que en Infoguía, el archivo generado se crea aproximadamente a la 1 de la madrugada del día siguiente al que pertenecen los datos, es decir, los datos del día 02/08/2012 serán los que tengan de nombre *Access.201208020000*. Así pues, igual que sucedía anteriormente, no siempre tienen el mismo nombre, hay ocasiones que las últimas 4 posiciones del nombre

del archivo varían. Pudiendo acabar en 0001 o 0030. En su momento se indagó acerca de los motivos de estas variaciones pero desde el departamento de tecnología no obtuvimos respuesta alguna.

#### **4.3.1 Sistema anterior**

En el procesado de datos de Infoguía, usábamos la herramienta FileZilla para realizar la conexión con el servidor y poder copiar en local los archivos necesarios. Eran 14 archivos, 7 por servidor, uno para cada día de la semana.

#### **4.3.2 Alternativas**

Para la recolección de datos podemos pensar otras opciones. Una de las más apropiadas es un script de copiado automático, que se realiza por línea de comandos. Sin necesidad de utilizar otro programa, solamente la consola CMD. Esto requiere una programación nueva, con su tiempo de desarrollo, sus pruebas y su implantación. Cuando queríamos desarrollar este sistema, nos encontramos con varios problemas: los archivos no siempre se guardan a la misma hora, ni con el mismo nombre, lo que hace muy difícil la selección automática de los mismos. Al variar el nombre, y no corresponderse siempre con la fecha, podríamos tener muchos fallos a la hora de procesar esos datos, que serían erróneos. Es por esto, que esta opción no salió a la luz. Además, teníamos la oposición por parte del departamento de seguridad. No estaba permitido el uso de scripts automáticos que no crean ellos mismos siempre que afecten a alguno de los servidores que alojan información relevante. Por estas razones, tuvimos que declinar cualquier tipo de script de copiado.

Aunque pueda parecer una excepción, es bastante común que los archivos estén cambiados de nombre, o que por algún error no se hayan generado. En tal caso, será necesario reclamarlos al departamento de sistemas. La concurrencia de alguno de estos errores era de aproximadamente el 10% de los archivos. Es decir, cada semana, de media, aproximadamente 1 archivo estaba o mal escrito o no aparecía.

#### **4.3.3 Solución Final**

La solución adoptada fue la misma que teníamos en Infoguía. Seguimos usando FileZilla, aplicación de libre distribución aprobada por el departamento de seguridad, y soportada por el departamento de sistemas. Simplemente tuvimos que configurar los accesos al servidor, por medio de su dirección, y la clave de acceso que nos proporciona el equipo de seguridad. Con las conexiones ya configuradas, simplemente debemos arrastrar los archivos como si se tratara de un explorador de Windows corriente, para poder trabajar con los archivos en local.

Estamos tratando archivos de 50MB los días de diario, o de 10MB aproximados los días festivos, esto varía por el hecho de que los agentes reciban más o menos llamadas a lo largo del día, y por lo tanto usen más o menos

Alejandría. Esta tarea llevará aproximadamente 10 minutos. Sin embargo, la persona encargada de realizar esta tarea puede dejar el ordenador funcionando, sin necesidad de estar delante. Es un proceso muy sencillo, que no suele fallar, por lo tanto no necesita supervisión. Lo que reduce el tiempo de trabajo dedicado a 5 minutos estimados. Lo que tardamos en conectar con el servidor, y arrastrar los archivos, que están colocados en una carpeta específica, a otra carpeta en local.

#### 4.4 Procesado de datos

Los logs recogidos del servidor pueden tener diferentes formatos. En este caso, nos pusimos en contacto con el departamento de sistemas, para que configuraran la salida de los mismos con la siguiente estructura: Fecha, URL y respuesta del servidor. Como hemos observado, los datos, al estar separados por comas, resulta muy sencillo tratarlos.

Lo que pretendemos en este bloque, es agrupar los datos para que cada dirección web tenga una entrada, con el número de visitas de la página. Al sumar las visitas a una web,

##### 4.4.1 Sistema anterior

Como ya hemos visto, en el sistema anterior a Analiz el agrupado de datos se realiza con una macro, que separa los enlaces correctos de los que han generado una respuesta errónea en el servidor. El resultado son 2 archivos con extensión 'txt', que recogen, una a una, las entradas con la fecha y dirección web. Además, una vez procesados los archivos con esta primera macro, era necesario procesar los ficheros resultantes con otra macro creada en Business Object. De esta segunda macro obteníamos los archivos categorizados que posteriormente mostraríamos en el cuadro de mando.

##### 4.4.2 Alternativas

La alternativa más clara que podíamos haber escogido para sustituir el método anterior es incluir esta parte de código en el siguiente apartado. De forma que primero procesáramos los datos, y posteriormente los subiéramos de una sola vez. Sin embargo, incluir el código ya existente en la programación PHP requería un tiempo añadido del que no disponíamos. A lo largo del proyecto observamos como los tiempos de entrega son ajustados, y en este caso, el mayor motivo para no haber adaptado esta parte del código es precisamente esa falta de tiempo.

Cualquier otra alternativa, que fuera procesar los datos de otra forma, o en otro lenguaje no tendría sentido. Las opciones eran claras, o mantenemos lo que ya hay, adaptándolo al nuevo sistema para que funcione, o lo incluimos directamente en el siguiente paso que damos, en programación PHP. Por motivos ya mencionados, se eligió adaptar la macro a la nueva plataforma.

#### 4.4.3 Solución Final

La solución adoptada, fue la de optimizar el procesado de datos que teníamos anteriormente. Así pues, partimos de la base de la macro que ya había creada. La primera modificación que necesitamos hacer, fue la que se encarga de reconocer la web de la que proceden los clics. En este caso, hemos analizado el *string* que contiene la dirección web, y comprobado que todas las direcciones tienen la palabra Alejandría en el lugar que deseamos. Esta comprobación se hace necesaria, para evitar coger datos de la antigua Infoguía, que también queda alojada en los mismos servidores, y sin embargo, esta información no queremos tratarla.

Otro de los cambios que hemos realizado, es que, la macro actual no une los datos de los dos servidores. La fuente, siguen siendo los 14 archivos que copiamos en la fase anterior del proceso, sin embargo, en este caso hemos preferido mantener dos archivos de salida, cada uno procedente de cada servidor. De forma que tratamos los archivos de uno de los servidores primero, y posteriormente del otro servidor. Esto es debido a que, después de hacer pruebas, nos hemos dado cuenta de que a la hora de subir los datos a la base de datos, daba menos fallos, al ser archivos tan grandes. De esta forma, el proceso de subida deberá duplicarse, sin embargo, hemos conseguido librar de errores esta fase. Por lo tanto, la salida tras realizar el procesado de datos, serán 2 archivos, en extensión 'csv' de un peso aproximado de 50mb.

#### 4.5 Almacenamiento de datos

El siguiente paso que debemos analizar es el del almacenamiento de los datos obtenidos. Es necesario guardar toda la información procesada, para su posterior visionado. Además, debemos mantener dicha información disponible hasta pasado un año, a partir de entonces empezará a formar parte de un archivo histórico. De esta forma, en caso de error percibido posteriormente, o si se desea volver a procesar los datos por cualquier cuestión, los tenemos disponibles al instante.

##### 4.5.1 Sistema anterior

En el anterior sistema nos encontrábamos con la información obtenida en el mismo medio en el que realizábamos la visualización de los datos, es decir, en la propia hoja de Excel se almacenaba semana tras semana los datos que necesitábamos. Así mismo, recurriamos a esta misma hoja para consultar cualquier dato que necesitáramos. Esto hace que las hojas de Excel que manejábamos dispusieran de mucha información, y fueran de difícil lectura. Así mismo, para copiar los datos obtenidos anteriormente al cuadro de mando, usábamos una macro, semi-automática. La calificamos de semi-automática porque si bien es cierto que solo era necesario pulsar 2 botones, e introducir el número de semana para que funcionara, muchas veces fallaba. Fallaba cada vez que se incluía una sección nueva en Infoguía, o cuando se suprimía. Se usaba un

sistema, que recurría a la búsqueda de un número en el archivo antes generado, y trataba de machearlo con el cuadro de mando. Si había cambiado alguno de los parámetros en el cuadro de mando, incluyendo columnas, o el archivo incluía más información de la prevista ,era necesario ajustar la macro manualmente. Esto sucedía cada semana, lo que resultaba un proceso hartamente difícil, y que requería demasiado tiempo de trabajo. Una vez estos datos estaban descuadrados, todo lo relativo a esa semana en el cuadro de mando, eran datos erróneos.

#### 4.5.2 Alternativas

Teniendo cómo premisa que queríamos huir de un cuadro de mando complejo, teníamos claro que no podíamos recurrir de nuevo a Excel. Así pues, una de las principales alternativas que se plantearon fue Access. Queremos almacenar los datos en una base de datos, para luego realizar consultas contra ella y poder obtener los valores que necesitamos. Access es una herramienta lo suficientemente potente para realizar esta tarea, además tenemos la ventaja de que está instalada en todos los equipos, por lo tanto las tareas de mantenimiento o gestión pueden realizarse en cualquier puesto de trabajo. Sin embargo, había poca gente en el departamento que supiera manejar Access a alto nivel. Además, poniendo la vista en el siguiente paso, era más difícil comunicar Access con una página web, que la opción que elegimos: SQL[5].

Nadie en el equipo sabía utilizar SQL, pero después de investigar un poco llegamos a la conclusión de que SQL era una tecnología más novedosa, con una gran extensión y aceptación en todo el mundo. Además la sencilla instalación y comunicación con el entorno web lo hacía ideal para nuestro proyecto.

Teniendo en cuenta esto, procedemos a analizar las distintas soluciones capaces de gestionar MySQL[6]. Cómo, además, sabemos que vamos a utilizar programación en PHP, dado que se relaciona de forma muy sencilla con MySQL, recurrimos a las aplicaciones 'AMP', estas son, todas las que agrupan Apache, MySQL y Php.

Este tipo de aplicaciones crean un servidor Apache que puedes ejecutar en tu ordenador, incluyendo la gestión de MySQL, y la integración de PHP, por esto, los programas AMP, son la solución más sencilla y completa.

Buscando entre los más destacados, encontramos : EasyPhp[7], XAMP[8], LAMP[9].

Para comenzar descartamos LAMP, principalmente, está estructurada para ser multiplataforma, y en nuestro caso, sólo la vamos a necesitar, hacer funcionar en Windows, por lo tanto no nos es necesario esta cualidad.

Entre las opciones restantes, EasyPhp y XAMP son bastante similares. Ambos entornos son de sencilla instalación, de fácil configuración y contienen ambos las últimas versiones de los 3 programas antes mencionados. Sin embargo EasyPhp

requiere menos carga de proceso en el ordenador, y es más fácil de adaptar a futuras versiones de Windows, como el 7 por ejemplo. Por esto mismo, decidimos elegir EasyPhp frente a XAMP.

### 4.5.3 Solución final

Finalmente nos decantamos por la creación de una base de datos SQL. Para ello recurrimos a la herramienta de libre distribución, que agrupa varios servicios, EasyPhp.

EasyPhp despliega un entorno 'ready-to-use' para desarrollar páginas en PHP. Instala un servidor interno en el PC, de forma que puedes usarlo para crear tus páginas web en PHP y combinarlas con las otras herramientas que vienen incorporadas.

Para ser más rigurosos, la versión que usamos de EasyPhp fue la 12.0. Dicha versión incorpora:

- PHP 5.2.17
- Apache 2.4.2
- MySQL 5.5.25
- phpMyAdmin 3.5.1
- Xdebug 2.2.0

De estas 5 herramientas, vamos a usar PHP, MYSQL y phpMyAdmin. El resto, a pesar de ser herramientas muy completas, no nos serán útiles en el desarrollo del proyecto.

PHP lo explicaremos en el siguiente apartado. Ahora trataremos el gestor de bases de datos MySQL.

MySQL es un sistema de gestión de bases de datos con 3 características muy interesantes:

- Relacional: los datos tienen relación, creando así subconjunto de datos. El orden de inclusión no importa, dado que posteriormente, mediante sentencias, puedes seleccionar el conjunto de datos que necesitas. Esta característica es básica a la hora de rastrear datos.
- Multihilo: esta propiedad nos permite lanzar varias acciones distintas a la vez. En esta ocasión no hemos usado dicha característica, pero su gran potencial podría ser útil en futuras mejoras.
- Multiusuario: podemos gestionar varias cuentas de usuarios, gestionando permisos y perfiles para cada uno de ellos.

Al margen de las características antes detalladas, hay que tener en cuenta, que MySQL se ha convertido en la base de datos por excelencia en el desarrollo web. A día de hoy es la herramienta más extendida para el almacenamiento y gestión de datos, por su facilidad de uso, su flexibilidad y ágil gestión de la información



contenida. Adicionalmente, conocemos su compatibilidad con la programación en PHP, lo que hace el conjunto perfecto para el proyecto a desarrollar.

phpMyAdmin es la herramienta que gestiona de forma visual el contenido de MySQL. Desde phpMyAdmin podemos crear bases de datos, tablas, dar permisos a usuarios de una forma muy sencilla y rápida. Se ha usado para dotar de una estructura lógica a la base de datos. Además hemos creado el perfil 'olealdi' que será el que tenga permisos para las conexiones desde PHP.

En la siguiente figura podemos observar el aspecto de phpMyAdmin.

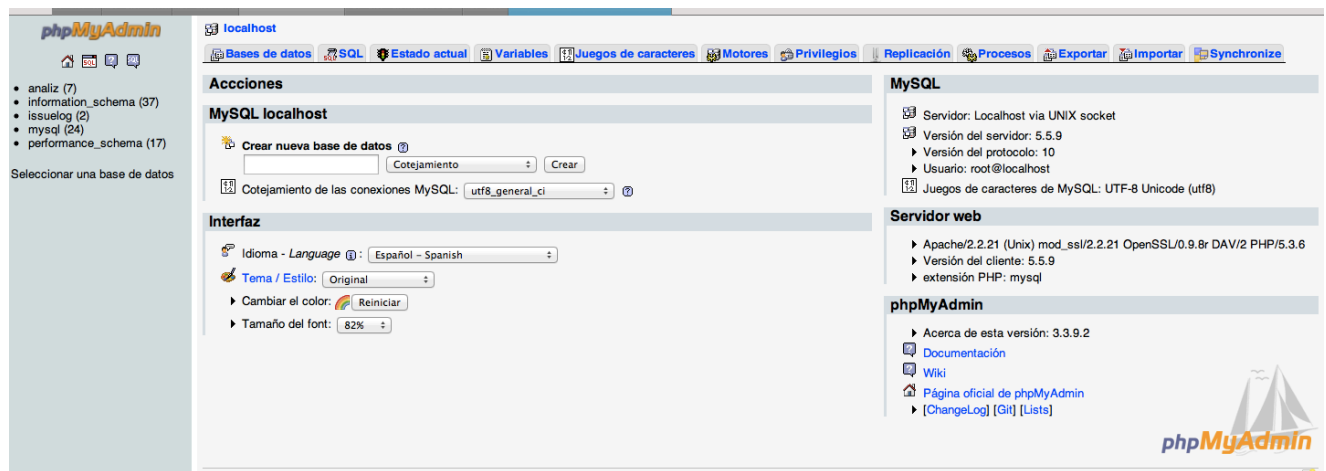


Figura 4.1 phpMyAdmin

Así pues, elegida la herramienta, solo queda darle una estructura la base de datos para que sea capaz de acoger los datos que queremos mandar. La base de datos que hemos creado se llama Analiz, como la herramienta de gestión de datos creada. Dentro de la base de datos disponemos de varias tablas.

De forma muy visual, en la siguiente imagen tenemos la descripción de agrupación de datos que hacemos en las bases de datos.



Figura 4.2 Agrupación de logs

En log\_Alejandría disponemos de los logs uno a uno, sin agrupar, y con la siguiente información:

- Número[int 18]: auto numérico que se incrementa automáticamente en cada entrada de nuevo dato, sirve para llevar un control. Limitamos el campo a 18 caracteres.

- Time [bigint 20]: fecha y hora en la que se generó esa entrada Limitamos el máximo a 20 caracteres.
- URL [varchar 230]: dirección web completa. Limitamos el máximo de la URL a 230 caracteres.
- Week [int 3]: semana a la que pertenecen los datos.
- Month [int 2]: mes al que pertenece la el acceso.
- Servicio [varchar 200]: código de servicio a la que pertenece la el acceso. Limitamos el máximo a 200 caracteres.
- Responses [int 4]: respuesta del servidor generada
- Memo [varchar 80]: etiqueta que indica el tipo de página a la que pertenece. Los valores que introducimos en esta categoría definen el tipo de URL a la que accedemos, por ejemplo: home principal, subhome, proceso, etc. Limitamos el máximo a 80 caracteres.

En el primer paso obtenemos una copia del 'csv' que hemos subido, en la tabla log\_Alejandria. En esta copia, al margen de los datos que nos ofrecen los logs, ya hemos realizado el análisis de URL, y tenemos mucha más información. Cómo hemos visto, la estructura de la tabla contiene numerosos campos para llevar un control más exhaustivo de cada acceso.

Posteriormente, realizamos una consulta contra la tabla log\_Alejandria agrupando los datos por 'URL' y sumando el número de veces que aparece dicha URL. De esta forma vamos rellenando la tabal Alejandría. Lo único que debemos añadir adicionalmente a la información que viene con cada URL, es el número de veces que ha encontrado dicha URL. Además, debemos despreciar la información de 'Time', dado que no tiene sentido almacenarla, al tratarse de agrupación de datos, se habrá accedido a la misma URL a diferentes horas de diferentes días, por lo tanto ese campo en un agregado no tiene cabida. Por otro lado, en esta tabla ya hemos filtrado las respuestas correctas del servidor, por lo tanto, el campo 'Response' tampoco tiene sentido mantenerlo.

Adicionalmente, rellenamos la tabla Errores\_Alejandría, con la misma información de cada entrada que Alejandría, pero en este caso sí mantenemos el campo 'Response', para conocer el fallo concreto del acceso a URL.

En la siguiente figura podemos observar la estructura de cada tabla utilizada.

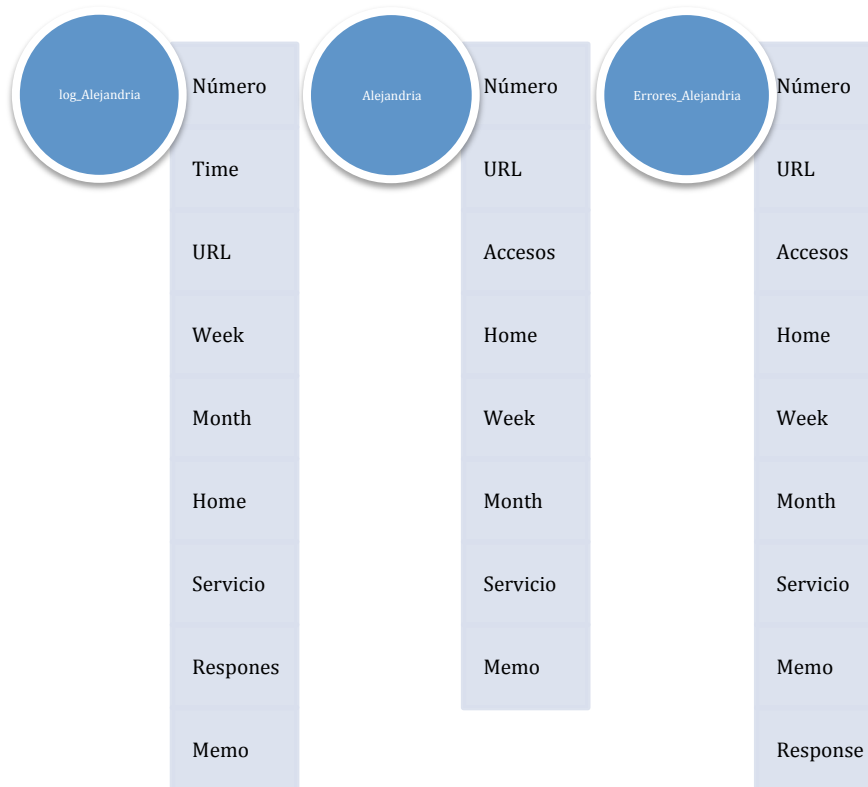


Figura 4.3 Estructura de tablas de Analiz

Todas las conexiones a la base de datos, tanto para subir datos nuevos, como para realizar consultas, están realizadas desde PHP. Primero creamos la conexión con la base de datos, y posteriormente elegimos la base de datos. Una vez realizado esto, podemos lanzar las consultas que necesitamos.

*Ejemplo de conexión:*

```
$conexion = mysql_connect("localhost", "olealdi", "olealdi");
mysql_select_db("analiz", $conexion);
```

## 4.6 Visualización de datos

La visualización de datos es una de las partes claves del proyecto en particular, y de suma importancia para el análisis de Alejandría. Una herramienta, aparte de ser potente, debe parecer simple, si no conseguimos esto, generará rechazo por los usuarios y no conseguiremos que la herramienta tenga aceptación, por lo tanto fracasará.

Por estas razones, y siguiendo una de las bases de Vodafone, 'simplicity', se decidió reducir al mínimo la cantidad de información lanzada al usuario a la vez, sin embargo no se cortaron las posibilidades de ahondar al máximo en la información necesaria.

#### 4.6.1 Sistema anterior

La visualización del sistema anterior se basaba en Excel. Se trata de una libro, con varias hojas, que muestra información exhaustiva de cada uno de los apartados visitados en Infoguía. Da información muy detallada de los accesos a la herramienta, segmentando la información hasta un nivel demasiado bajo, que nadie va a necesitar consultar. En su día, se planteó el sistema, de forma que el cuadro de mando sirviera de reporting al mismo tiempo que de almacenado de datos. Por esto mismo podemos encontrarnos en el libro diferentes hojas, unas con gráficas, otras con datos puros.

Como podemos ver en la siguiente figura, la información mostrada es abrumadora. Es cierto que podemos encontrar datos de casi cualquier acceso que necesitemos, empezando por la Home principal, pasando por cada una de las Homes de los diferentes departamentos, paginas intermedias y agrupaciones de productos. Sin embargo, la visualización que hacemos de los datos es simultanea, llegando a resultar una vista demasiado compleja para los departamentos ajenos al nuestro.

#### 4.6.2 Alternativas

Partiendo de la base de que la elección de la tecnología ya viene predefinida por la selección de la base de datos, es decir, como hemos escogido MySQL, que se relaciona a la perfección con PHP, tenemos que construir un entorno web que sirva tanto de herramienta para gestionar, como de herramienta de visualización. Esto nos deja las posibilidades muy claras: debemos programar en PHP un entorno web, para luego poder relacionar la base de datos con dicha web.

Por otro lado, debemos pensar la mejor forma para visualizar la información, no solo para llegar a ella, sino para ver el dato concreto que queremos obtener. En este punto la decisión estaba clara: necesitamos gráficas. Las gráficas representan la información de forma concisa y visual, de forma que impacta más en el usuario y tiene mejor aceptación. Además es más rápido de entender que una tabla, y mucho más intuitiva.

Así pues, hicimos una búsqueda entre las posibilidades de representar gráficas mediante PHP. Encontramos varias opciones, jPGraph[10] fue una de ellas. Esta librería nos sirve para representar todo tipo de gráficas de modo simple. El usuario solo debe encargarse de cargar los datos de manera correcta, y esta librería se encarga de pintarlos. Sin embargo, tiene una punto negativo importante: no es vistoso. Buscamos una librería que impacte en el usuario, a ser posible con gráficos dinámicos e interactivos, y en este punto, encontramos una solución perfecta: FusionCharts[11].

### 4.6.3 Solución final

Para analizar la solución de visualización escogida, haremos hincapié en la interfaz que manejamos para llegar a los datos, esto es, el entorno web. Y posteriormente comentaremos la forma de visualización de los datos en sí por medio de FusionCharts.

#### 4.6.3.1 Entorno Web

Tanto para la gestión de administrador de almacenado de datos, como para la consulta de los mismos, se ha programado en PHP/HTML un entorno web desde el que podemos acceder a todas las opciones que son necesarias con el uso normal de Analiz.

Se ha recurrido a PHP por estar orientado al desarrollo de aplicaciones web dinámicas con información almacenada en bases de datos. Como vemos, la definición encaja perfectamente con el tipo de programación que buscamos. Además, es un lenguaje sencillo y seguro, dado que el código escrito en PHP es invisible para el navegador y para el cliente, ya que es el servidor el que se encarga de ejecutarlo y mandar el resultado al navegador.

Para hacernos a la idea del entorno creado, podemos visualizar los siguientes pantallazos, a modo de ejemplo de navegación que pueden resultar útiles.

En la siguiente figura podemos ver la pantalla de inicio del gestor de aplicaciones realizado. Desde ella tenemos acceso a las tres aplicaciones diseñadas en el transcurso de la beca de colaboración.

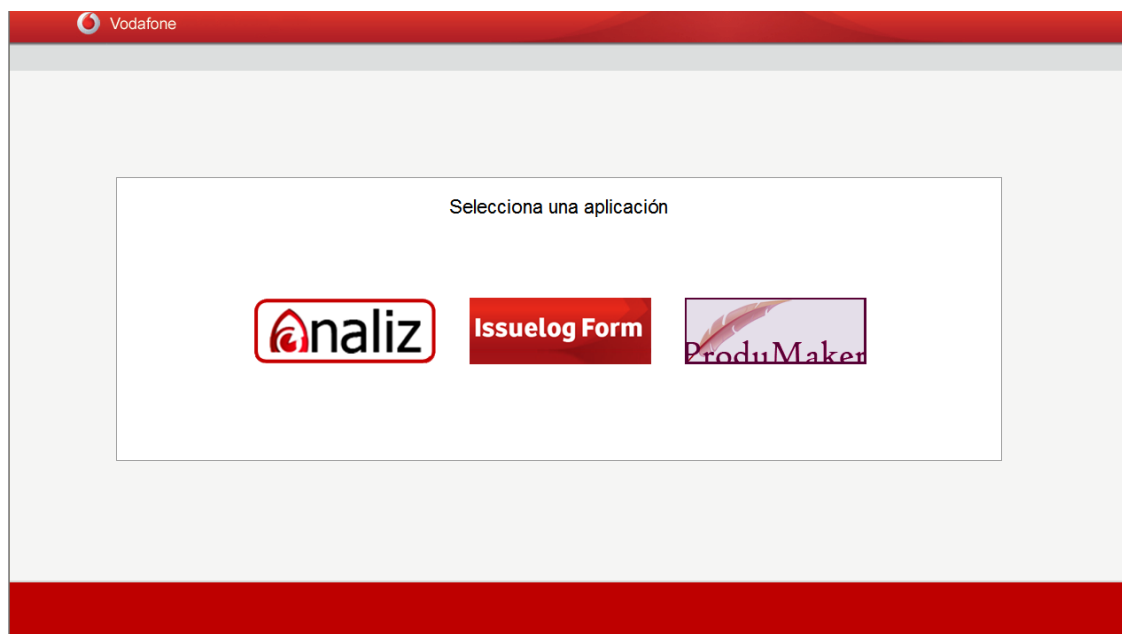


Figura 4.4 Gestor de aplicaciones

Si hacemos click en Analiz, vamos a la página de inicio de la aplicación. Como vemos, tenemos dos opciones: Consultar Datos y Panel de Administración.

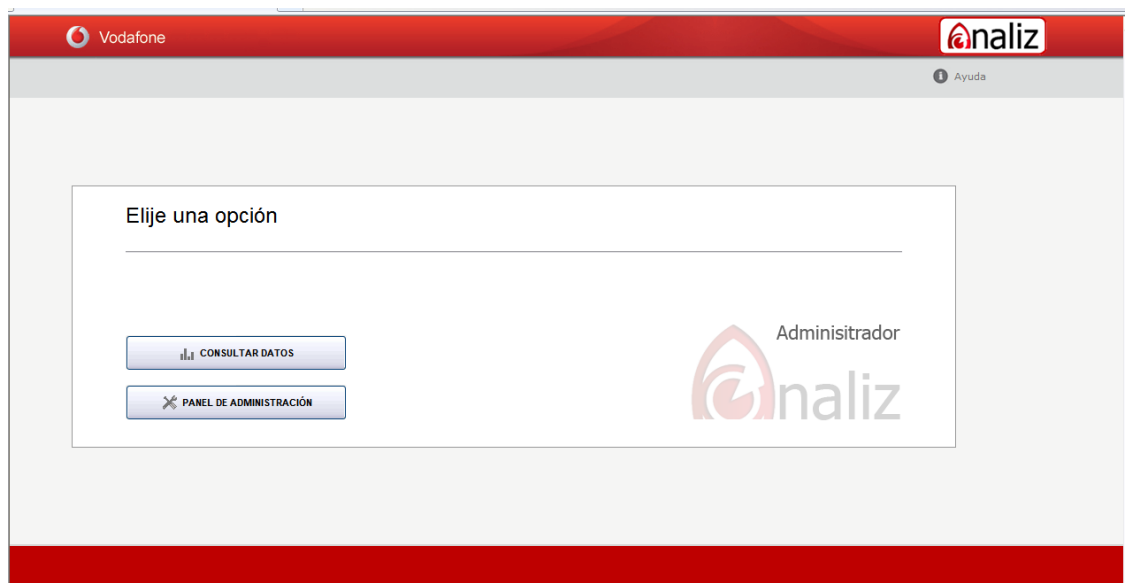


Figura 4.5 Analiz

Desde Consultar Datos, accedemos a las opciones que tenemos abiertas para los usuarios: Todas las consultas de datos.

En la opción de panel de administración, tenemos las opciones abiertas al equipo de formación, donde podemos realizar todas las tareas de administración:

- **Subida de datos:** desde esta opción podremos subir los datos relativos a una semana. Procesando los logs, para pasarlos a la base de datos, y posteriormente poder consultar las estadísticas.
- **Subida de valoraciones:** esta opción nos permite actualizar las estadísticas de valoraciones que se recogen en Analiz.
- **Introducir nueva categoría:** esta opción se introduce por si surge un cambio de estructura en Alejandría. Mediante ella podemos añadir una nueva categoría que relaciona las subhomes con los códigos que se usan en la dirección URL para poder fragmentarla.
- **Borrado log\_Alejandria:** borra la tabla auxiliar de log\_alejandría. Esa opción está implementada para liberar de memoria la base de datos.

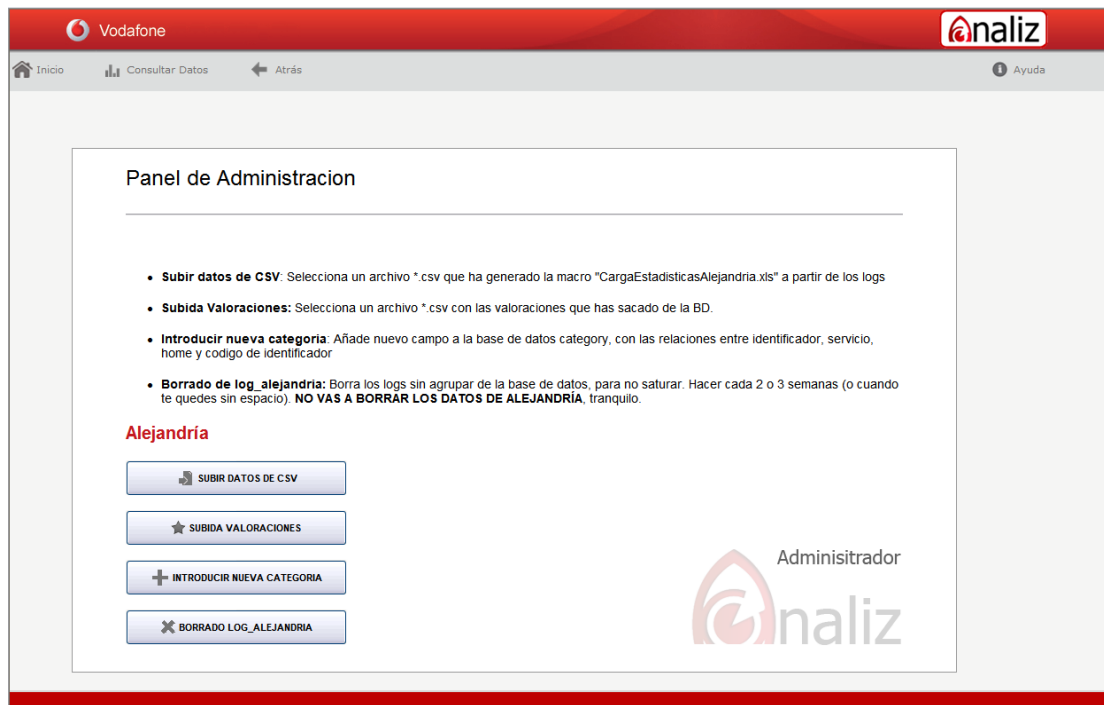


Figura 4.6 Panel de administración

En la opción de consultar datos podemos encontrar toda la parte que aporta usabilidad a la herramienta. En ella implementamos todas las búsquedas de datos que recurren a la base de datos, y muestran la información solicitada.

Como podemos observar en la siguiente figura, podemos acceder a diferente información:

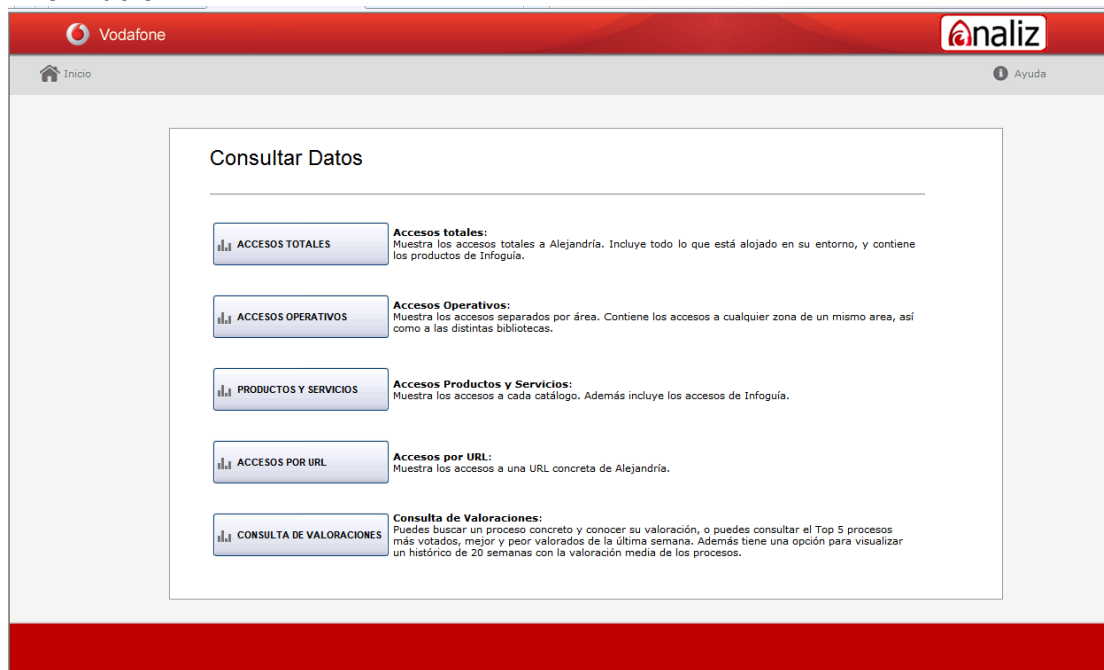


Figura 4.7 Analiz consultas

- **Accesos Totales:** Muestra los accesos totales a Alejandría. La suma de visitas a todo el entorno de Alejandría. Además, aparece información acerca de las visitas a procesos, y visitas a subhomes. Adicionalmente, podemos encontrar la información relativa a las visitas a catálogos, a bibliotecas y a archivos de biblioteca. Toda esta información la da distribuida en dos graficas, y una tabla. La información mostrada, aparece a lo largo de las últimas 6 semanas, si queremos consultar información más antigua, podemos recurrir al botón “Más datos” de forma que nos aparece la misma información, pero con una antigüedad de 28 semanas, para obtener una visión global del acceso a Alejandría.

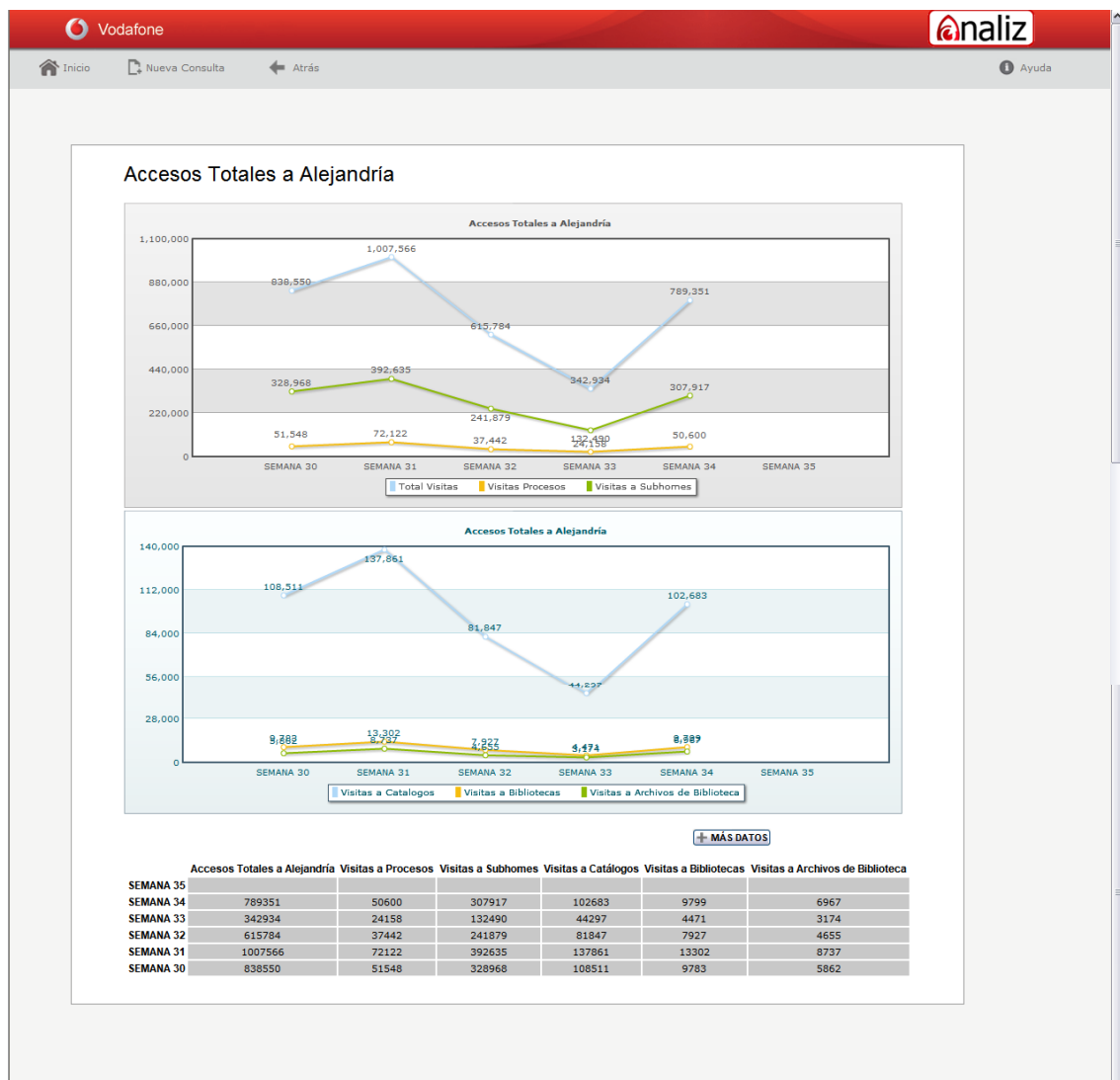


Figura 4.8 Analiz accesos totales

- **Accesos Operativos:** Esta opción muestra la suma de accesos a subhome y procesos pertenecientes a dicho servicio. Por medio de una selección, podremos seleccionar el servicio del que queremos obtener información, y del mismo modo que en la consulta general, obtendremos información de las últimas 6 semanas.



- **Productos y Servicios:** Desde esta opción podremos obtener los datos relativos a los accesos a cada catalogo (Pospago, prepago, autónomos y empresas), con datos relativos a las últimas 6 semanas.

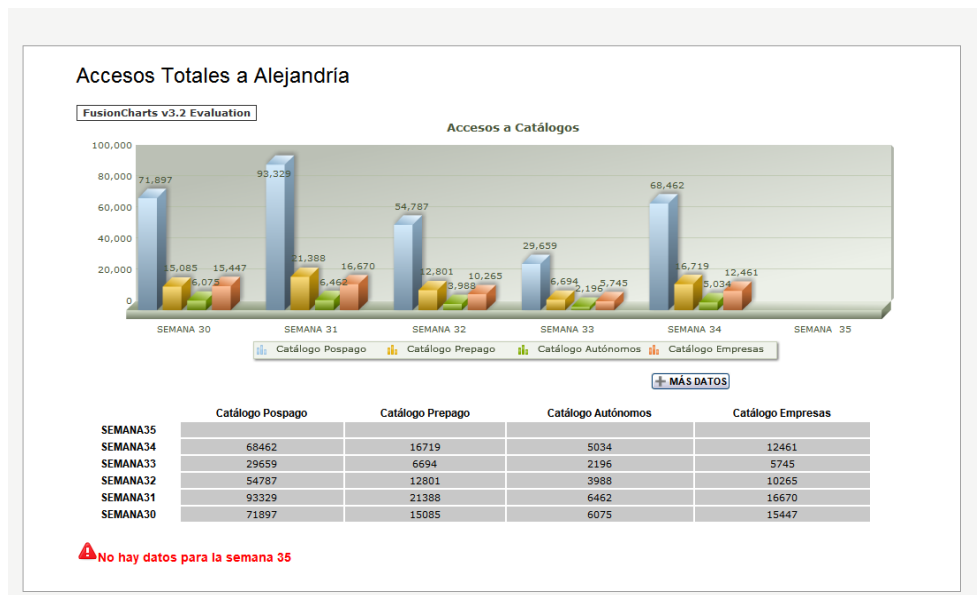


Figura 4.9 Analiz accesos a catálogos

- **Accesos por URL:** Esta opción se incluye para cubrir la necesidad de consultar una URL concreta. Simplemente debemos copiar y pegar la URL, y pulsar el botón 'buscar'. Obtendremos el número de accesos a esa web concreta con accesos de las últimas 6 semanas. Del mismo modo que en las opciones anteriores, tenemos un histórico de 28 semanas.
- **Consulta de valoraciones:** Podemos consultar la valoración de un proceso concreto, o consultar el volumen total de valoraciones de las últimas 6 semanas. También podemos consultar el número de procesos distintos valorados, y la media de las valoraciones de los procesos. Además, podemos consultar cuales fueron los procesos más valorados, los mejor valorados, y los peor valorados de la última semana. Cómo cabe esperar, también tenemos información relativa a las 28 últimas semanas recurriendo al histórico.

Durante toda la navegación entre páginas, tenemos una cabecera con opciones que nos permiten navegar más fácilmente, cómo podemos observar en la siguiente figura.



Figura 4.10 Analiz Cabecera

Desde las opciones que nos ofrecen, podemos ir a la página de inicio, que es la página inicial donde podremos elegir consultar datos, o gestionar la

herramienta. Realizar una nueva consulta, o simplemente retroceder una página en el historial.

Adicionalmente, tenemos un apartado de ayuda, desde el que explicamos el modo de uso de la herramienta, así como informamos de todos los datos que pueden obtener mediante su uso.

#### 4.6.3.2 FusionCharts

FusionCharts es una herramienta que ayuda a crear gráficos interactivos para páginas web. Tiene una librería con un amplio número de gráficos, basados en JavaScript y Flash. Gracias a esto, consigue una impresión muy atractiva al usuario y una experiencia de uso inmejorable. Está especialmente orientada a XML, sin embargo, es fácilmente adaptable a PHP. Por estos motivos escogemos esta herramienta para la visualización de nuestras gráficas.

Tiene un punto negativo, y es que no se trata de una herramienta de libre distribución, es necesario pagar la licencia. Sin embargo tenemos la suerte de que cuenta con una versión de prueba, que será la que pongamos en marcha para Analiz. El departamento está recortando gastos, y lo único que diferencia la versión de prueba de lo que no lo es, en nuestro caso, es que aparece el nombre de la herramienta en una esquina del gráfico. Asumiendo este detalle, la herramienta tuvo una gran acogida en el departamento.

La ventaja de usar FusionCharts, a parte de lo vistosa que es, es que es prácticamente automática. Simplemente has de definir los parámetros, y mandar la información, y la herramienta ajusta sus límites y crea sus relaciones para mostrar un gráfico dinámico e interactivo. El hecho de que sea interactivo nos permite seleccionar en tiempo real la información que estamos viendo, accediendo al valor concreto de un punto, u ocultando series de datos que no queremos visualizar.

Por ejemplo, en la siguiente figura, podríamos ocultar todas las barras de una serie simplemente haciendo clic en la leyenda que aparece en la parte inferior.

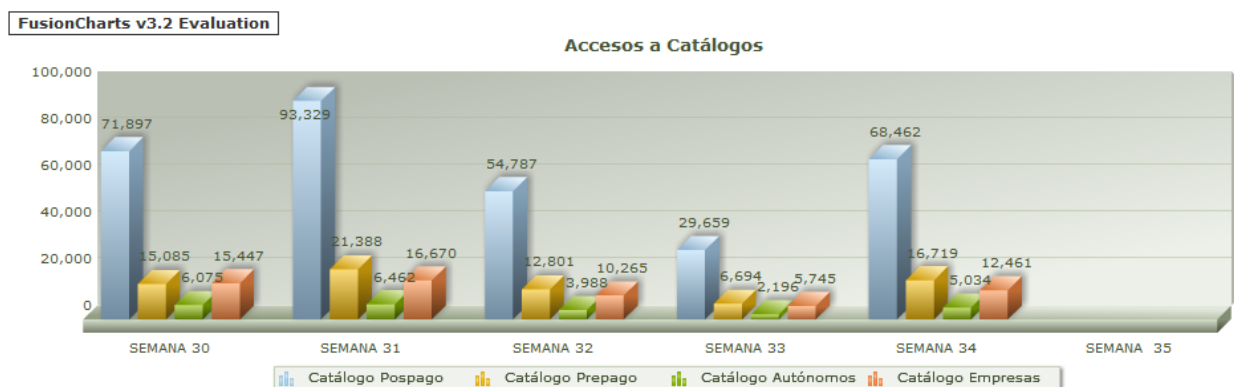


Figura 4.11 Ejemplo FusionCharts

Cómo se ha comentado anteriormente, es totalmente parametrizable, pudiendo elegir desde los colores de las barras o el fondo, hasta la orientación de los valores, incluso podemos ocultarlos o cambiar el tamaño de los mismos. Todo esto, son atributos que acepta FusionCharts en la definición del gráfico. A continuación podemos ver el fragmento de código que implementa el gráfico anterior mostrado.

```
<embed src="../../graphs/MSColumn3D.swf"
FlashVars="&chartWidth=500&chartHeight=400&debugMode=0&dataXML=
<chart baseFontSize='10' showValues='1' formatNumberScale='0'
alternateHGridAlpha='20' divLineAlpha='50' caption = 'Accesos a Cat-logos'
baseFontSize='10' showValues='1' formatNumberScale='0'>
```

*<--! Definimos el número de semana almacenado anteriormente -->*

```
<categories>
<category label = 'SEMANA <?php echo $myweek[5]; ?>' />
<category label = 'SEMANA <?php echo $myweek[4]; ?>' />
<category label = 'SEMANA <?php echo $myweek[3]; ?>' />
<category label = 'SEMANA <?php echo $myweek[2]; ?>' />
<category label = 'SEMANA <?php echo $myweek[1]; ?>' />
<category label = 'SEMANA <?php echo $myweek[0]; ?>' />
</categories>
```

*<--! Definimos las diferentes series, recogiendo los valores almacenados de la consulta -->*

```
<dataset seriesName='Catalogo Pospago' >
<?php
$j=5;
while ($j>=0){
echo "<set value =\"".$fila1[$j].\" />";
$j--;
}?>
</dataset>
<dataset seriesName='Catalogo Prepago' >
<?php
$j=5;
while ($j>=0){
echo "<set value =\"".$fila2[$j].\" />";
$j--;
}?>
</dataset>
```

*[...]*

```
</chart>" quality="high" width="940" height="300"
allowScriptAccess="always" type="application/x-shockwave-flash"
pluginspage="http://www.macromedia.com/go/getflashplayer" />
```

```
</object>
```

Como hemos observado, debemos tener los datos bien almacenados, previa consulta a la base de datos, para posteriormente inyectarlos en las propiedades de FusionCharts para que los muestre.

#### 4.7 Documentación

Una vez concluido el proyecto, es necesario crear la documentación. En este caso tendremos 2 documentos claves para el proyecto.

El primero de ellos es la memoria del proyecto. Este documento, el que están leyendo, se estructura y desarrolla de forma posterior al proyecto en sí. Recoge toda la información necesaria para su entendimiento, así como la justificación de métodos elegidos.

El otro documento queda a disposición del departamento de formación y aprendizaje de Vodafone. Se trata de un manual de usuario. En él aportamos toda la información precisa para que cualquier otra persona que llegue al departamento sea capaz de hacer funcionar, modificar o sustituir alguna parte del sistema. De esta forma, el proyecto no depende de una sola persona, sino que cualquiera puede realizar las tareas de mantenimiento y gestión necesarias.

El manual de usuario explica de forma concisa los puntos más conflictivos de la aplicación así como la forma de actuar en caso de encontrar algún problema. Así mismo, describe la estructura de datos creada para Analiz, de forma que sea entendible a ojos ajenos al proyecto. Además, todo el código de la aplicación está correctamente indentado y comentado para facilitar las tareas de mantenimiento.

## Capítulo 5. Implantación y pruebas

### 5.1 Implantación

El proyecto se ha estado desarrollando desde el principio en la máquina que posteriormente se iba a usar como servidor interno, con la única restricción de que durante la implantación no se ha publicado la dirección del mismo, y se han cortado los accesos. De esta forma podíamos tener un implantación sin sobresaltos, en un entorno de trabajo controlado y con unos tiempos de actuación definidos.

Así pues, se eligió la máquina del departamento que está dedicada a las labores de procesado y almacenado de estadísticas. Al tener un equipo dedicado, nadie debe ceder su puesto a la hora de la obtención de estadísticas.

Las primeras pruebas reales se hicieron coincidentes con la semana de despliegue de Alejandría. De esta forma conseguimos una imagen alineada con de los informes de reporting con la situación del departamento.

### 5.2 Pruebas iniciales

Como toda herramienta con un desarrollo medianamente complejo, Analiz requería el lanzamiento de unas pruebas iniciales. Con ellas, comprobamos los posibles fallos que tuviera la herramienta, y así subsanarlos antes de que siguiera adelante el proyecto.

Así pues, las primeras pruebas se realizaron, como he comentado anteriormente, a la vez que las primeras pruebas de Alejandría. Demasiados interrogantes en el escenario, y demasiadas posibilidades de fallo al estrenar dos herramientas a la vez.

Como cabría esperar, se encontraron los primeros fallos a la hora de trabajar con datos reales.

En la siguiente tabla podemos encontrar los fallos y sus soluciones:

PROBLEMA	PROGRAMA/ENTORNO	SOLUCIÓN
Permisos para la obtención de datos de los servidores remotos	FILEZILLA – OBTENCIÓN DE DATOS	Hubo que tramitar una solicitud de acceso desde el equipo donde alojamos nuestro servidor. Seguridad tardó 3 días en darnos acceso
Falta de archivos logs en los servidores	OBTENCIÓN DE DATOS	Hubo falta de algunos archivos lo que nos imposibilitó el procesado de datos.

<b>Formato erróneo en los logs salientes del servidor</b>	<b>PROCESADO DE DATOS</b>	<p>Contactamos con tecnología y fue imposible recuperarlos.</p> <p>Contactamos con el equipo de tecnología para que adaptaran el formato de salida de los logs, al que teníamos anteriormente. Tardaron 1 día en procesar la petición.</p>
<b>Subida de archivos a la base de datos</b>	<b>MYSQL – ALMACENADO DE DATOS</b>	<p>Al ponerse en marcha Alejandría, los archivos que vienen del servidor tienen un peso mucho mayor del esperado. Al procesarlos e intentar subirlos, tenemos un fallo del servidor, por agotar el tiempo de espera. Lo solucionamos configurando EasyPhp con un mayor tiempo de espera.</p>

**Tabla 5.1 Errores y solución**

Una vez solucionados los problemas iniciales, la aplicación se despliega de forma interna. Se abre al departamento, para que se pruebe desde varios equipos y se compruebe el funcionamiento, y lo que es más importante: se ponga a prueba la interfaz del sistema.

Una de las características que pueden hacer que un proyecto fracase o triunfe es su interfaz. Por ello la interfaz de Analiz está cuidada, es simple, limpia, para que cualquier persona que vaya usarlo, pueda hacerlo sin necesidad de recurrir al manual de usuario.

Tras corregir algunas descripciones y comprobar que todo funcionaba correctamente, se pone a disposición de todos los departamentos, para que puedan realizar sus consultas sin necesidad de que desde dentro del departamento tengan que facilitar ese dato.

## Capítulo 6. Planificación y Presupuesto

---

### 6.1 Planificación

En el inicio del proyecto se fijó una planificación que se ha ido cumpliendo a lo largo del mismo. Se fijaron fechas para un desarrollo faseado del mismo, para que de esta forma fuera más sencillo su progreso y poder realizar el seguimiento de forma más precisa.

La fecha de inicio del proyecto fue el día 18 de Julio de 2011, con la primera reunión se sentaron las bases de lo que se pretendía hacer.

El primer paso fue crear una planificación que se puede visualizar en la siguiente figura.

El análisis del sistema anterior llevó un total de 10 jornadas. Con este análisis pretendíamos introducirnos en el mundo del sistema anterior, conocer sus puntos débiles y fuertes, para poder basarnos en ellos a la hora de desarrollar el nuevo sistema.

El estudio de alternativas fue la clave para el desarrollo del proyecto. En esta fase se sentaron las bases para la realización del mismo. Con una duración de 8 jornadas, se recurrió a búsquedas en internet para documentarnos acerca de todas las posibilidades que podíamos valorar.

En la implementación de Analiz empleamos 38 jornadas. Creamos el sistema completo, aunque la parte más compleja fue la que dota de una estructura a la base de datos, así como la creación de la web que permite la visualización de los datos.

Posteriormente se pasa a realizar las primeras pruebas, en ellas detectamos errores y pequeñas modificaciones a realizar. Tras corregir esto, realizamos una segunda fase de pruebas, y con el aprobado del departamento, llegamos a la fase de despliegue. En esta última fase abrimos los accesos a la web y comunicamos la herramienta como novedad, de forma que los usuarios pudieran acceder y comprobar el resultado del proyecto.

A continuación podemos ver el esquema de tareas con fechas de organización, y el diagrama de planificación del proyecto.

Task Name	Duration	Start	Finish	Predecessors
<b>Planificación</b>	2	07/18/11	07/19/11	
Planificación	2	07/18/11	07/19/11	
<b>Análisis sistema anterior</b>	10	07/20/11	08/02/11	
Recolección de datos	2	07/20/11	07/21/11	2
Procesado de datos	3	07/22/11	07/26/11	4
Copiado a cuadro de mando	3	07/27/11	07/29/11	5
Almacenado de histórico	2	08/01/11	08/02/11	6
<b>Estudio de alternativas</b>	8	08/03/11	08/12/11	
Estudio de alternativas	8	08/03/11	08/12/11	7
<b>Implementación Analiz</b>	38	08/15/11	10/05/11	
Recolección de datos	3	08/15/11	08/17/11	9
Procesado de datos	5	08/18/11	08/24/11	11
Creación de base de datos	15	08/25/11	09/14/11	12
Creación entorno web	15	09/15/11	10/05/11	13
<b>Primeras Pruebas</b>	7	10/06/11	10/14/11	
Primeras Pruebas	7	10/06/11	10/14/11	14
<b>Corrección errores</b>	7	10/17/11	10/25/11	
Corrección errores	7	10/17/11	10/25/11	16
<b>Pruebas definitivas</b>	5	10/26/11	11/01/11	
Pruebas definitivas	5	10/26/11	11/01/11	18
<b>Despliegue aplicación</b>	4	11/02/11	11/07/11	
Despliegue aplicación	4	11/02/11	11/07/11	20

Figura 6.1 Esquema de tareas



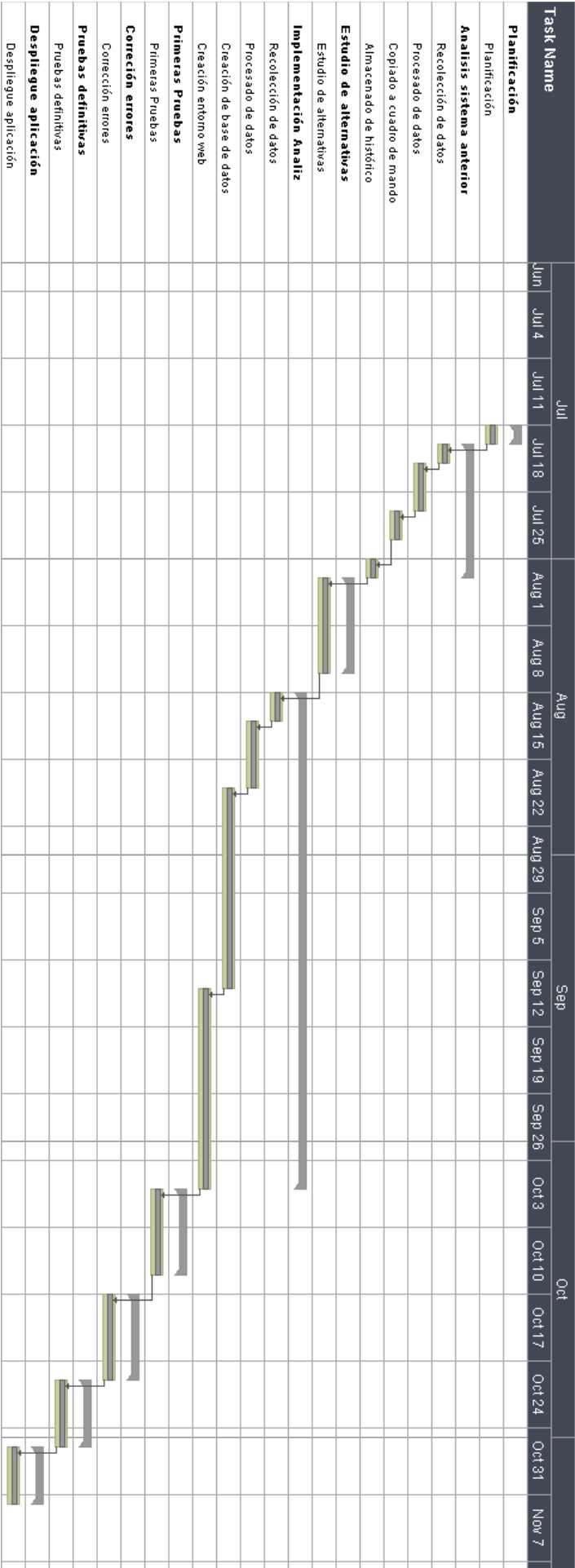


Figura 6.2 Diagrama de planificación

## 6.2 Presupuesto

A continuación desglosaremos el presupuesto según las categorías de costes de personal, software y hardware, para posteriormente ofrecer un cuadro resumen con el coste total del proyecto.

### 6.2.1 Costes de personal

Los costes de recursos humanos relacionados con el proyecto están vinculados directamente con la planificación del mismo. No tendremos en cuenta los horarios laborales, simplemente contabilizaremos horas y personal dedicado al desarrollo del trabajo.

Así mismo, definiremos la jornada laboral asociada a la beca como:

1 Trabajador mes (105 horas) = 5 horas/día x 5 días x 4,2 semanas.

Categoría	Nº Horas	Coste hora	Coste Total
Jefe de proyecto	20	75 €	1.500 €
Analista	100	6,25 €	625 €
Diseñador	75	6,25 €	468,75 €
Programador	170	6,25 €	1062,5 €
Pruebas Software	55	6,25 €	343,75 €
TOTAL			4.000 €

Tabla 6.1 Coste Personal

### 6.2.2 Costes de software

A lo largo del proyecto, tanto para crearlo como para gestionarlo e implantarlo, hemos utilizado diferentes herramientas con licencia. En la siguiente tabla podemos apreciar el coste imputable del proyecto en relación a estas herramientas.

Para calcular los costes estimamos el periodo de amortización de 5 años para software. Para estimar el coste asociado a dicho equipamiento se ha hecho uso de la siguiente fórmula:

$$\text{Coste} = \left(\frac{A}{B}\right) \times C \times D$$

Donde A: número de meses desde la fecha de facturación en que el equipo es utilizado

B: periodo de depreciación

C: coste del equipo (sin IVA)

D: porcentaje de uso que se dedica al proyecto. Lo fijamos al 100%

En la siguiente tabla podemos ver los costes asociados al proyecto:

Descripción	Coste	Dedicación (meses)	Periodo de depreciación	Coste Imputable
Microsoft Windows XP	300 €	4	60	20 €
Microsoft Office 2003	470 €	4	60	31,3 €
FileZilla	0 €	4	60	0 €
EasyPhp	0 €	4	60	0 €
FusionCharts	0 €	4	60	0 €
Notepad ++	0 €	4	60	0 €
Gimp	0 €	4	60	0 €
<b>TOTAL</b>				<b>51,3 €</b>

Tabla 6.2 Coste Software

### 6.2.3 Coste Hardware

Del mismo modo que hemos hecho con los elementos software, haremos ahora con todos los componentes hardware que involucra el proyecto.

Descripción	Coste	Dedicación (meses)	Periodo de depreciación	Coste Imputable
PC Sobremesa	499 €	4	60	33,26 €
Monitor LCD 22 pulgadas HP v220	129 €	4	60	8,6 €
Ratón y Teclado HP	23 €	4	60	1,53 €
<b>TOTAL</b>				<b>43,49 €</b>

Tabla 6.3 Coste Hardware

### 6.2.4 Resumen de costes

A continuación mostraremos el resumen de costes, incluyendo las tablas anteriormente vistas. Además incluiremos un 20% de costes indirectos que será el presupuesto final alcanzado mediante el uso de la plantilla del presupuesto. Así mismo hemos incluido una segunda tabla añadiendo un 18% como concepto de impuestos de valor añadido.

CONCEPTO	VALOR
Coste Personal	4.000 €
Coste Software	51,3 €
Coste Hardware	43,49 €
Materia Fungible	30 €

Gastos Generales (20 % coste personal)	800€
<b>Subtotal 1</b>	<b>4.924,79 €</b>
Beneficios Empresariales (20 % subtotal 1)	984,96 €
<b>Subtotal 2</b>	<b>5.909,75 €</b>
<b>Coste del riesgo (10 % Subtotal 2)</b>	<b>590,97 €</b>
<b>Base Imponible</b>	<b>6.500,72 €</b>
<b>IVA (18% Base Imponible)</b>	<b>1.170,13 €</b>
<b>TOTAL</b>	<b>7.670,85 €</b>

Tabla 6.3 Resumen de costes

## Capítulo 7. Líneas Futuras y Conclusiones

---

### 7.1 Líneas Futuras

El proyecto tiene muchas posibilidades de evolución. Al tratarse de un proyecto en empresa cabe destacar la presión a la que se es sometido para que el trabajo vea la luz. Hay un tiempo específico, una carga de trabajo asignada y una planificación que se debe respetar. Es por esto que muchas de las novedades que nos hubiera gustado implementar, no ha sido posible hacerlo.

#### 7.1.1 Automatización

Uno de los puntos más interesantes sería la automatización del proyecto completo. Para cualquier empresa es fundamental el tiempo y la carga de trabajo que recibe un trabajador, por ello, esta línea de trabajo tiene mucho sentido.

Podríamos conseguir la automatización de la obtención de archivos. Mediante un script de copiado, y una conexión ftp protegida, podríamos acceder a los archivos que se generan en los servidores y copiarlos en local para poder trabajar con ellos. En éste sentido, uno de los problemas con los que nos encontraríamos serían los cortafuegos de seguridad, sin embargo con la petición pertinente podría llegar a autorizarse.

A parte de esto, tendríamos el problema de que los archivos no siempre se nombran igual, y desde luego, nunca se generan a la misma hora. Habría que generar una horquilla de posibilidades en las cuales se encajarían nuestros posibles archivos, susceptibles de ser copiados.

Una vez tuviéramos todos los archivos en local, no sería demasiado complejo iniciar un script capaz de comenzar a procesar los datos. Sin embargo, sería necesario un cambio de tecnología, dejar de usar una macro de Excel para usar otro tipo de gestor que hiciera lo propio. El objetivo del posible cambio de tecnología es que cualquier programación será más fácil de implementar que involucrar al programa Excel.

Posteriormente, habría que ejecutar todo el código que ya hay creado, para almacenar las estadísticas en la base de datos y poder visualizarlas.

#### 7.1.2 Visionado de estadísticas

Cómo hemos visto en el proyecto, a la hora de visionar estadísticas, se usan *queries* ya predefinidas. Si bien es cierto que a petición de cada área se podrían ir incluyendo otras consultas más específicas, esto requeriría un soporte técnico que trabajara detrás. En nuestro afán por automatizar, podríamos encontrar una solución de búsqueda de datos que permitiera seleccionar, con la máxima flexibilidad posible, la consulta que se quisiera realizar, añadiendo o quitando datos según fuera necesario, y haciendo lo propio con el tiempo de observación,

es decir, si queremos datos de aquí a un mes, o solamente necesitamos la última semana.

Sería similar a implementar una especie de tabla dinámica dentro de Analiz. Del mismo modo que hace Excel, te calcula en tiempo real el dato que necesitas, realizando una conjunción de los valores que se ha elegido mostrar. Esto mismo se podría conseguir por medio de un formulario, para que sea más sencillo, y configurar una página de estadísticas personalizada.

### 7.1.3 Perfiles

En la línea de trabajo anterior, podemos incluir una gestión de usuarios: lo normal es que una persona siempre consulte los mismos datos.

Al añadir una opción de usuarios, podríamos conservar las preferencias de cada usuario. De esta forma, cuando alguien se logara con su nombre de usuario y contraseña, tendría sus *queries* favoritas guardadas. Y por tanto sus estadísticas saldrían siempre con el mismo aspecto, lo que lo haría mucho más versátil para incluirlas en informes futuros.

Además, con esta opción de los perfiles, podríamos tener diferentes posibilidades de permisos. Poniendo varios niveles: usuario, administrador. De esta forma limitaríamos el acceso de una mejor forma, a cómo está actualmente.

### 7.1.4 Optimización

Así pues, se podría trabajar en la optimización de los sistemas. Éste es un punto muy genérico que puede aplicar a cualquier proyecto, sin embargo podemos focalizarlo en diferentes aspectos:

- **Procesado de datos:** cómo hemos visto, el tiempo aproximado que requiere el hecho de unificar los archivos provenientes de los servidores para reducirlos y compilarlos en uno solo es de 20 minutos, aproximadamente. Convendría ser capaces de reducir este tiempo, bien mejorando los algoritmos que se usa para éste procesado, o bien invirtiendo dinero en una máquina nueva más potente.
- **Base de datos:** de la misma forma que en el paso anterior, sería primordial reducir el tiempo que tardan los datos en almacenarse en la base de datos. Mejorar este aspecto es sin duda una línea de trabajo futura clave.

## 7.2 Conclusiones

A lo largo de la carrera se han ido planteando diferentes retos que he logrado superar, retos que me han preparado para mentalizarme a la hora de enfrentarme a un nuevo proyecto. En este caso, no solo he tenido la oportunidad de poner en práctica la mentalidad con la que se nos prepara durante la carrera,

además he podido participar en la dinámica de empresa que nos espera a la salida de la universidad.

A lo largo del proyecto he conseguido obtener un amplio conocimiento de programación PHP, así como de gestión de base de datos MySQL. Si bien he aplicado los conceptos adquiridos tanto en forma de pensar para la programación, como en la gestión de MySQL adquirida en una de las últimas asignaturas, también puedo asegurar que no he dejado de adquirir conocimientos a lo largo de todo el proyecto.

Durante la carrera hemos cursado diferentes asignaturas de programación en distintos lenguajes, situación que creo me ha sido especialmente útil en la adaptación a PHP. Así mismo, hemos tenido libertad suficiente para poder gestionar diferentes respuestas frente a un mismo problema, lo que nos ha abierto la mente para poder analizar las diferentes soluciones posibles y compararlas objetivamente para adoptar la óptima.

Gracias a esto, y a la consecución del proyecto en sí, conseguir una herramienta de visualización de estadísticas completa, podemos concluir que se han cumplido los objetivos tanto laborales como personales ampliamente.

La realización de un proyecto para una empresa como Vodafone ha sido una oportunidad de suma importancia para mi crecimiento personal y laboral.

El hecho de tratarse de la primera incursión en el mundo profesional hace de este proyecto algo más especial, si cabe. Adaptarse a la forma de trabajo de una empresa, asumiendo responsabilidades poco a poco, y cogiendo peso laboralmente hablando es una experiencia única que me ha servido para madurar en muchos aspectos de mi vida.

Las empresas trabajan con poco margen de maniobra. La sobrecarga de los trabajadores y los cortos plazos de entrega muchas veces merman las posibilidades de un producto. Sin embargo, la presión en algunos momentos del proyecto ha sido claves para lograr una resolución favorable.

Queda la experiencia aprendida, las horas dedicadas a un trabajo que ha visto luz. Analiz está actualmente siendo usada por el departamento de formación y aprendizaje. Esto, para mí, es un importante reconocimiento al mérito de un proyecto que me ha llevado gran parte del año.

Meses después de la finalización del proyecto, he tenido la oportunidad de volver a Vodafone y he podido comprobar que siguen usando Analiz como fuente de datos. Por todo esto, y después de la experiencia vivida, valoro muy positivamente tanto el tiempo de beca, como la realización del proyecto.

## Capítulo 8. Bibliografía

---

- [1] About Management KPI: Key Performance Indicators [07/07/2012]  
<http://management.about.com/cs/generalmanagement/a/keyperfindic.htm>
- [2] Clientes Vodafone [08/07/2012]  
<http://www.vodafone.es/conocenos/es/vodafone-espana/quienes-somos/historia/>
- [3] Manual usuario de Infoguía/Alejandría (recurso impreso propiedad de Vodafone) [08/07/2012]
- [4] FileZilla webpage [08/07/2012]  
<http://filezilla-project.org>
- [5] SQL official webpage [08/07/2012]  
<http://www.sql.org>
- [6] MySQL official webpage [08/07/2012]  
<http://www.mysql.com>
- [7] EasyPhp web page, forum and faq [08/07/2012]  
<http://www.easyphp.org>
- [8] XAMP information [08/07/2012]  
<http://www.apachefriends.org/en/xampp.html>
- [9] LAMP installation guide [08/07/2012]  
<http://www.lamphowto.com>
- [10] JpGraph information [08/07/2012]  
<http://jpgraph.net>
- [11] FusionCharts documentation and examples [08/07/2012]  
<http://www.fusioncharts.com>